



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

# **Análise de Modelos de Representação de Informação Normativa Usando XML**

Felipe Camargos Costa

Monografia apresentada como requisito parcial  
para conclusão do Bacharelado em Ciência da Computação

Orientador

Prof. Dr. Francisco de Assis Cartaxo Pinheiro

Brasília  
2017



# Análise de Modelos de Representação de Informação Normativa Usando XML

Felipe Camargos Costa

Monografia apresentada como requisito parcial  
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Francisco de Assis Cartaxo Pinheiro (Orientador)  
CIC/UnB

Prof. Dr. Díbio Leandro Borges      Prof. Dr. Rodrigo Bonifácio de Almeida  
CIC/UnB      CIC/UnB

Prof. Dr. Rodrigo Bonifácio de Almeida  
Coordenador do Bacharelado em Ciência da Computação

Brasília, 4 de julho de 2017

# Resumo

Esta monografia estuda as seguintes propostas de padronização, usando XML, de documentos normativos, legislativos e parlamentares: Akoma Ntoso, CEN Metalex, USLM e LexML Brasil. A linguagem XML possui características que a tornam adequada como solução para a construção deste tipo de padronização, além de ser amplamente usada, com várias ferramentas já construídas para o tratamento de documentos escritos nesta linguagem. Esta e outras razões são abordadas nesta monografia. Além disto, é feita uma análise do panorama das propostas de padronização, estudando suas características, soluções adotadas, contexto de criação e implantação da solução e diferenças e semelhanças entre as propostas.

**Palavras-chave:** XML, Linguagens de marcação de documentos, representação de documentos normativos, documentos parlamentares, OASIS, Akoma Ntoso, LexML Brasil

# Abstract

This monograph examines the following standardization proposals, using XML, of normative, legislative and parliamentary documents: Akoma Ntoso, CEN Metalex, USLM and LexML Brasil. The XML language has characteristics that make it suitable as a solution for the construction of this type of standardization, in addition to being widely used, with several tools already built for the treatment of documents written in this language. This and other reasons are addressed in this monograph. In addition, an analysis of the standardization proposals is analyzed, studying their characteristics, solutions adopted, context of creation and implementation of the solution, and differences and similarities between the proposals.

**Keywords:** XML, Markup languages, Representation of legal texts, parliamentary documents, OASIS, Akoma Ntoso, LexML Brasil

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Problema . . . . .	1
1.2	Objetivos . . . . .	2
1.3	Metodologia e Estrutura . . . . .	2
<b>2</b>	<b>Fundamentos da linguagem XML</b>	<b>3</b>
2.1	O que é XML . . . . .	3
2.2	Histórico da XML . . . . .	4
2.3	Características de XML . . . . .	6
2.4	Estrutura de um documento XML . . . . .	7
2.4.1	Elementos XML . . . . .	8
2.4.2	Nomes . . . . .	9
2.4.3	Declarações . . . . .	9
2.4.4	Entidades não processadas . . . . .	11
2.4.5	Seções CDATA . . . . .	11
2.4.6	Instruções de processamento . . . . .	11
2.4.7	Comentários . . . . .	12
2.5	Espaços de nomeação . . . . .	12
2.6	Verificando e validando documentos XML . . . . .	12
2.6.1	Verificação . . . . .	13
2.6.2	Validação baseada em tipos de documento . . . . .	13
2.6.3	Validação baseada em esquemas XML . . . . .	13
2.7	Transformando documentos XML . . . . .	14
2.7.1	Apresentação . . . . .	14
2.7.2	Transformando documentos XML . . . . .	14
2.8	Programação com documentos XML . . . . .	15
2.8.1	Processamento sequencial . . . . .	16
2.8.2	Processamento baseado em árvore . . . . .	16
2.8.3	Outros modos de acesso . . . . .	16

2.9	Schema XML . . . . .	17
2.10	URI, URL e URN . . . . .	17
2.10.1	URIs . . . . .	17
2.10.2	URLs . . . . .	17
2.10.3	URNs . . . . .	18
<b>3</b>	<b>A Importância do Problema</b>	<b>19</b>
3.1	Necessidade de modelar textos legais . . . . .	19
3.2	Propostas para modelagem de textos legais . . . . .	21
<b>4</b>	<b>Soluções Apresentadas Existentes</b>	<b>24</b>
4.1	CEN Metalex . . . . .	24
4.2	Akoma Ntoso . . . . .	25
4.3	USLM . . . . .	27
4.4	LexML Brasil . . . . .	29
<b>5</b>	<b>Análise Crítica</b>	<b>33</b>
5.1	O que foi proposto . . . . .	33
5.2	O que foi obtido . . . . .	33
5.3	Pontos semelhantes entre as propostas . . . . .	33
5.4	Pontos ou abordagens divergentes . . . . .	35
<b>6</b>	<b>Conclusão</b>	<b>36</b>
6.1	Revisão do panorama atual . . . . .	36
6.2	Propostas de trabalhos futuros . . . . .	37
	<b>Referências</b>	<b>38</b>
	<b>Apêndice</b>	<b>40</b>
<b>A</b>	<b>Exemplo - Lei de São Salvador</b>	<b>41</b>
A.1	Ley 604 de 1993 da República de El Salvador . . . . .	41
A.2	Folha de estilo para a Ley 604/93 . . . . .	50

# Lista de Figuras

2.1	Diagrama de Venn para URIs, URNs e URLs [16]. . . . .	18
4.1	Exemplo de URN do LexML Brasil [13]. . . . .	32
A.1	Exemplo de saída produzida para a Ley 604/93 de El Salvador. . . . .	51

# Lista de Tabelas

2.1 Elementos de um documento XML. . . . .	8
--	---



# Capítulo 1

## Introdução

O movimento para criar uma padronização que permita as interconexões entre os vários textos normativos dentro de um mesmo país, e entre diferentes países, está em suas fases iniciais.

A codificação de textos normativos em XML é reconhecidamente importante. Muitos países e organizações estão propondo e desenvolvendo padrões para esta finalidade. Esses padrões, além de atenderem as necessidades locais das comunidades para as quais foram concebidas, procuram manter um conjunto mínimo de elementos compatíveis (a compatibilidade no ambiente globalizado é fundamental).

A padronização de um formato de representação eletrônico de textos legais pode permitir o uso de ferramentas computacionais para obter informação desses textos, bem como facilitar a construção de novos textos e a sua distribuição.

### 1.1 Problema

No Brasil a codificação legal de textos normativos ainda se encontra em fase inicial não consolidada, assim como a maioria dos países, existindo iniciativas dispersas. Diferentes sítios podem representar leis de maneiras distintas. Como por exemplo, pode-se citar o sítio da Câmara dos Deputados [1] e o sítio do Planalto [2].

É necessário avaliar até que ponto um padrão já existente pode ser usado para representar os textos legais do ordenamento jurídico brasileiro, isto é, decidir se este padrão é capaz de suprir as necessidades encontradas no Brasil pelos diversos agentes envolvidos na produção, utilização e distribuição do código legal.

A padronização da representação de código normativo, aliada ao acesso aberto de documentos governamentais, está intimamente ligada a transformações na relação de governos com a sociedade, na medida que promove transparência e fornece informação para que a sociedade possa participar ativamente das discussões e políticas de seus governan-

tes, efetivamente favorecendo a democratização e a eficiência da administração pública. Algumas iniciativas foram criadas para atingir esse objetivo. Um exemplo é o Akoma Ntoso [22], criado pelas Nações Unidas no contexto de um projeto para o fortalecimento da democracia na África.

## 1.2 Objetivos

- Estudar algumas entre as diversas propostas existentes para padronização de código legal, em diferentes países.
- Analisar e discutir os resultados dessas representações e decidir sua adequação.

## 1.3 Metodologia e Estrutura

- O capítulo 2 fornece um embasamento sobre a linguagem XML, sua definição, histórico, estrutura e exemplos de uso são apresentados.
- O capítulo 3 apresenta o problema e discute a necessidade e a importância de um modelo de representação de informação normativa
- O capítulo 4 apresenta algumas iniciativas já criadas para a padronização da representação de código legal no formato XML. São elas: CEN Metalex, Akoma Ntoso, USLM e LexML Brasil.
- O capítulo 5 discute os pontos de convergência e as divergências entre as iniciativas citadas no capítulo 4.
- O capítulo 6 conclui o trabalho e sugere trabalhos futuros.

## Capítulo 2

# Fundamentos da linguagem XML

Este capítulo se destina a fazer uma exposição da XML, dada a sua importância na realização dos objetivos deste trabalho. A seção 2.1 apresenta uma definição informal da XML e ajuda a elucidar alguns dos principais conceitos relacionados. A seção 2.2 expõe as origens da XML, o contexto de sua criação, seus criadores e algumas tecnologias que a antecederam. A seção 2.3 apresenta a terminologia, os conceitos básicos, características e objetivos da XML. A seção 2.4 distingue um documento bem formado em XML dos demais e os aspectos estruturais do mesmo. A seção 2.5 explana *namespaces*. A seção 2.6 discorre sobre validação de um documento XML e apresenta conceitos para a construção de uma ferramenta de validação. A seção 2.7 informa como fazer transformação de um documento XML em outros formatos, tais como HTML e XHTML.

### 2.1 O que é XML

*XML* é a abreviação de *Extensible Markup Language*. Segundo o documento oficial de recomendação da XML, pelo W3C [27], é uma linguagem que descreve uma classe de objetos de dados chamados documentos XML e parcialmente descreve o comportamento de programas de computadores que os processam. XML é uma aplicação específica, ou uma forma mais restrita de SGML, abreviação de *Standard Generalized Markup Language* [ISO 8879] [18]. Por construção, documentos XML estão em conformidade com documentos SGML [27].

*Markup*, ou em português, *Marcação*, é a informação incluída num documento para melhorar seu significado de certas maneiras, por identificar as partes e como elas se relacionam umas com as outras. Para tanto, a marcação se utiliza de símbolos de marcação, ou *tags*, para fornecer rótulos e limites a um documento, de forma que partes do texto possam ser distinguíveis das outras, permitindo que um programa de computador possa dar tratamento adequado para cada elemento do texto, ao invés de tratar o documento

inteiro como um bloco de texto contíguo [26]. Desta forma, podemos dizer que a marcação torna um documento passível de compreensão por programas de computador.

Por sua vez, uma *linguagem de marcação* é um conjunto de símbolos que pode ser colocado no texto de um documento para demarcar e rotular as partes desse documento. A XML não é uma linguagem de marcação por si só, mas uma especificação, ou conjunto de regras, para criar linguagens de marcação. Isto ocorre porque, apesar de existirem algumas regras mínimas sobre como formar e usar *tags* em XML, quaisquer nomes de *tag* válidos podem ser usados e eles podem aparecer em qualquer ordem. Isso é semelhante a construir suas próprias palavras, observando as regras de pontuação. Quando um documento satisfaz às regras mínimas da XML, diz-se que ele é *bem formado* [26].

A XML está atualmente na versão 1.0 e sua especificação mais recente pode ser encontrada na página web que hospeda o documento de especificação da sintaxe XML do W3C [27]. XML segue um padrão aberto, o que significa que a XML não está ligada a uma empresa ou instituição isolada e nem acoplada a qualquer software em particular [26].

## 2.2 Histórico da XML

Os primeiros formatos eletrônicos de organização da informação se preocupavam mais em descrever como as coisas apareciam (apresentação) que com a estrutura e o significado do documento. *troff* e *TEX*, duas das primeiras linguagens de formatação, fizeram um trabalho fantástico para formatar documentos impressos, mas perderam qualquer senso de estrutura. Consequentemente, os documentos eram limitados a serem vistos na tela ou impressos em papel. Você não poderia escrever programas facilmente para pesquisar e canalizar informações, fazer referência cruzada eletronicamente ou redefinir a finalidade dos documentos para aplicações diferentes [26].

A *codificação genérica*, que usa *tags* descritivas em vez de códigos de formatação, por fim resolveu esse problema. A primeira organização a explorar essa idéia foi a *Graphic Communications Association* (GCA). No final da década de 1960, o projeto "*GenCode*" desenvolveu maneiras de codificar diferentes tipos de documento com *tags* genéricas e reunir documentos a partir de várias partes [26].

O próximo grande avanço foi a *Generalized Markup Language* (GML), um projeto da IBM. Os projetistas da GML, *Charles Goldfarb*, *Edward Mosher* e *Raymond Lorie*, a idealizaram como uma solução para a codificação de documentos para uso com vários subsistemas de informação. Os documentos codificados nessa linguagem de marcação poderiam ser editados, formatados e pesquisados por diferentes programas, devido às suas *tags* baseadas em conteúdo. A IBM, uma imensa editora de manuais técnicos, utilizou bastante a GML, provando a viabilidade da codificação genérica [26].

Inspirado pelo sucesso da GML, o *Committee on Information Processing* do *American National Standards Institute* (ANSI) montou uma equipe, tendo *Goldfarb* como líder de projeto, para desenvolver uma linguagem padrão de descrição de texto baseada na GML. O comitê GenCode da GCA também contribuiu com sua habilidade. Durante o final da década de 1970 e o início da década de 1980, a equipe publicou rascunhos de trabalho e por fim criou um candidato para um padrão do setor (GCA 101-1983), chamado *Standard Generalized Markup Language* (SGML). Essa linguagem foi rapidamente adotada pelo Departamento de Defesa e pela Receita Federal dos Estados Unidos [26].

Nos anos que se seguiram, a SGML realmente começou a decolar. O *International SGML User's Group* começou a se reunir no Reino Unido em 1985. Junto com a CGA, eles espalharam o evangelho da SGML por toda a Europa e a América do Norte. Estendendo a SGML em um âmbito maior, o *Electronic Manuscript Project* da *Association of American Publishers* (AAP) encorajou o uso da SGML para codificar documentos de uso geral como livros e jornais. O Departamento de Defesa dos Estados Unidos desenvolveu aplicações para SGML em seu grupo *Computer-Aided Acquisition and Logistic Support* (CALS), incluindo um tipo de documento popular para formatação de tabela, chamado *CALS Tables*. Depois, dando cobertura a esse início bem-sucedido, a *International Standards Organization* (ISO) ratificou um padrão para a SGML (ISO 8879:1986) [26].

A SGML foi projetada para ser um esquema de codificação flexível e abrangente. Como a XML, ela é basicamente um kit de ferramentas para o desenvolvimento de linguagens de marcação especializadas. Mas a SGML é muito maior que a XML, com uma sintaxe mais aberta e muitos parâmetros esotéricos. Ela é tão flexível que o software criado para processá-la é por demais complexo e caro, sendo sua utilidade limitada a grandes organizações, que possam pagar tanto pelo software quanto pelo custo da manutenção de complicados ambientes SGML [26].

A revolução pública na codificação genérica aconteceu no início da década de 1990, quando a *Hypertext Markup Language* (HTML) foi desenvolvida por Tim Berners-Lee e Anders Berghund, funcionários do laboratório europeu de física das partículas, CERN. O CERN esteve envolvido no esforço para a SGML desde o início da década de 1980, quando *Berghund* desenvolveu um sistema de publicação para testar a SGML. *Berners-Lee* e *Berghund* criaram um tipo de documento SGML, compacto e eficiente, para documentos de hipertexto. Foi fácil escrever software para essa linguagem de marcação, e mais fácil ainda codificar documentos. A HTML escapou do laboratório e prosseguiu para dominar o mundo [26].

No entanto, a HTML de certa forma era um passo atrás. Para conseguir a simplicidade necessária de modo a ser verdadeiramente útil, alguns princípios de codificação genérica tiveram que ser sacrificados. Por exemplo, um tipo de documento era usado para todas

as finalidades, forçando as pessoas a sobrecarregarem *tags* em vez de definirem *tags* de finalidade específica. Em segundo lugar, muitas das *tags* são puramente para apresentação. A estrutura simples dificultava saber onde uma seção começava e outra terminava. Muitos documentos codificados com HTML hoje se baseiam tanto na formatação pura, que não podem ter sua finalidade redefinida com facilidade. Apesar disso, a HTML foi um passo brilhante para a Web e um pulo gigante para as linguagens de marcação, pois fez com que o mundo se interessasse em documentação e vínculo eletrônico [26].

Para retornar aos ideais da codificação genérica, algumas pessoas tentaram adaptar a SGML para a Web - ou melhor, adaptar a Web à SGML. Isso provou ser muito difícil. A SGML era muito grande para ser colocada em um browser da Web pequeno. Era necessária uma linguagem menor, que ainda mantivesse a generalidade da SGML - assim nasceu a *Extensible Markup Language* (XML) [26].

Estimulado pela insatisfação com os formatos existentes (padronizados ou não), um grupo de empresas e organizações que se autodenominou *World Wide Web Consortium* (W3C) começou a trabalhar em meados da década de 1990 em uma linguagem de marcação que combinava a flexibilidade da SGML com a simplicidade da HTML. Sua filosofia na criação da XML era incorporada por vários princípios importantes [26].

Formatos eletrônicos anteriores foram mais propriamente projetados para descrever a visualização das coisas do que com a estrutura e o significado. Consequentemente, os documentos eram limitados a serem lidos na tela ou impressos em cópias físicas, Você não poderia escrever programas para procurar ou extrair informação, referenciar eletronicamente ou dar novos propósitos aos documentos para diferentes aplicações. Códigos genéricos, que usam tags descritivas ao invés de códigos formatados, eventualmente resolveram esse problema [26].

## 2.3 Características de XML

A XML é bastante exigente quando se trata de estrutura. Um documento deverá ser marcado de tal forma que não haja duas maneiras de interpretar os nomes, a ordem e a hierarquia dos elementos. Isso reduz bastante os erros e a complexidade do código. Os programas não precisam fazer um chute educado ou tentar consertar erros de sintaxe, como os *browsers* HTML fazem constantemente, pois não existem surpresas de um processador XML criando um resultado diferente de outro [26].

Naturalmente, isso torna mais difícil a escrita de uma boa marcação XML. Você precisa verificar a sintaxe do documento com um analisador, a fim de garantir que os programas usados mais adiante funcionarão sem erros, que a integridade dos seus dados está protegida e que os resultados serão coerentes [26].

Além da verificação básica da sintaxe, você pode criar suas próprias regras de como um documento deve se parecer. A DTD é um gabarito para a estrutura do documento. Um esquema XML pode restringir os tipos de dados permitidos dentro dos elementos (por exemplo, datas, números ou nomes). As possibilidades para verificação de erro e controle da estrutura são incríveis [26].

*XML* permite nomear as partes de tudo o que quiser, ao contrário da *HTML*, que o limita a nomes de *tags* predefinidos. Para a *XML*, não importa como você vai usar o documento, como aparecerá quando formatado ou ainda o que significam os nomes dos elementos. Tudo o que importa é que você siga as regras básicas para a marcação. Entretanto, isso não significa que questões de organização não sejam importantes. Você deverá escolher nomes de elemento que façam sentido no contexto do documento, em vez de coisas aleatórias, como os signos do zodíaco. Isso é para o seu próprio benefício (e daqueles que usam sua aplicação *XML*) do que qualquer outra coisa.

Os objetivos de projeto da XML são:

- XML deverá ser amplamente usada através da Internet.
- XML deverá suportar uma ampla variedade de aplicações.
- XML deverá ser compatível com SGML.
- Deverá ser fácil construir programas que processam documentos XML.
- O número de características opcionais em XML deve ser mantido no mínimo absoluto, idealmente zero.
- Documentos XML devem ser legíveis para humanos e razoavelmente claros.
- O projeto da XML deve ser preparado rapidamente.
- O projeto da XML deve ser formal e conciso.
- Documentos XML devem ser fáceis de criar.
- Concisão em marcação XML é de mínima importância.

## 2.4 Estrutura de um documento XML

Os documentos XML são organizados por meio de elementos que descrevem sua estrutura hierárquica e como devem ser processados. Há os elementos de estruturação, que podem ser vazios ou possuir conteúdo, as declarações (elementos declarativos), as instruções de processamento (elementos de execução), os comentários (elementos explicativos), as seções CDATA (elementos não interpretáveis) e as referências (elementos de referência).

Tabela 2.1: Elementos de um documento XML.

ELEMENTO	DESCRIÇÃO	EXEMPLO
vazio	Representa uma informação específica totalmente especificada por sua etiqueta.	<code>&lt;xref linkend="abc"/&gt;</code>
conteúdo	Representa uma região do documento que pode conter texto e elementos.	<code>&lt;p&gt;parágrafo&lt;/p&gt;</code>
declaração	Define um parâmetro, entidade ou aspecto gramatical para ser usado pelo analisador sintático	<code>&lt;!ENTITY author "Nome do autor"&gt;</code>
instrução	Usado para fornecer instrução a tipos particulares de aplicações	<code>&lt;?print-formatter force-linebreak?&gt;</code>
comentário	Inserir um comentário (ignorado pelo processador XML)	<code>&lt;!--texto explicativo--&gt;</code>
CDATA	Cria uma seção de dados (do tipo caractere) cujo conteúdo não é analisado	<code>&lt;![CDATA[Sinais e-comercial: &amp;&amp;&amp;&amp;&amp;]]&gt;</code>
referência	Refere-se a um texto armazenado em outro lugar, para que seja inserido no ponto de referência	<code>&amp;nome-organização;</code>

Cada elemento é designado por uma etiqueta que inicia com o caractere `<`, seguido do identificador da etiqueta, e termina com o caractere `>`, antecedido de um ou mais caracteres de finalização. Para os elementos que definem uma região do documento a etiqueta inicial deve vir acompanhada de uma etiqueta final, que geralmente tem seu identificador idêntico ao identificador da etiqueta que designa o início da região (antecedido do caractere `\`). A Tabela resumo Tabela 2.1 é baseada em [26]:

### 2.4.1 Elementos XML

Os elementos são a unidade estrutural mais comum em um documento XML. Eles podem ser vazios, apenas indicando a ocorrência de alguma informação em um dado ponto do documento para, por exemplo, permitir a inserção de uma figura nesse ponto; ou podem ser do tipo contêiner, isto é, possuir conteúdo que por sua vez pode consistir de elementos e texto.

Cada elemento é designado por uma etiqueta que inicia com o caractere `<`, seguido do identificador da etiqueta e termina com o caractere `>`. A etiqueta pode ter atributos. No exemplo a seguir, a etiqueta `período` possui dois atributos de nomes `ini` e `fim`:

```
<período ini='2010' fim="2016">
```



Os atributos de uma etiqueta são pares de nomes e valores ligados pelo caractere =, com os valores entre aspas simples ou duplas. Eles definem propriedades associadas à informação que a etiqueta descreve.

Se o elemento é do tipo contêiner, ele deve ser terminado com uma etiqueta finalizadora, sem atributos, cujo nome é idêntico ao da etiqueta inicial, antecedido do caractere /. O par de etiqueta designa o elemento do tipo contêiner e entre eles pode existir texto e outros elementos:

```
<endereço tipo="trabalho">
  Universidade de Brasília
  <dept>Ciência da Computação</dept>
  Campus Darcy Ribeiro
</endereço>
```

### 2.4.2 Nomes

Os nomes em XML são formados por caracteres alfanuméricos, incluindo caracteres de idiomas não latinos como cirílico, grego, japonês, chinês etc., e pelos símbolos de pontuação hífen (-), sublinhado (\_) e ponto (.). Nomes devem começar com uma letra, um ideograma ou um sublinhado.

Entre a chave angular e o nome que designa a etiqueta não pode haver espaços.

### 2.4.3 Declarações

A declaração XML é opcional, mas se estiver presente deve ser a primeira do prólogo. Seus parâmetros são opcionais, sendo o parâmetro **version** obrigatório, se outros estiverem presentes. Os demais são **encoding**, que define o conjunto de caracteres usado no documento, e **standalone**, que indica se todas as declarações estão contidas no próprio documento (valor **yes**) ou se há declarações fora do documento. A ordem dos parâmetros é requerida: primeiro a versão, seguida da codificação, seguida da autocontenção.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

### Declaração DOCTYPE

A declaração que define o tipo de documento pode ser usada para especificar entidades de valores padrões para os atributos, ou para permitir a validação do documento — um modo especial de verificação sintática que checa a gramática e o vocabulário usado na marcação do documento. Esse segundo uso da declaração faz-se com a indicação de um documento de definição de tipo (DTD, do inglês *Document Type Definition*).

```
<!DOCTYPE doc SYSTEM "/usr/local/xml/dtds/simple.dtd">
```

A indicação do documento DTD pode ser feita no modo **system** seguido da identificação do endereço local ou do endereço da internet (como um identificador URI) entre aspas. Também pode ser usado um identificador público, o qual supõe-se não poder ser modificado. Entretanto, na falta de um mecanismo universalmente aceito para a implementação de identificadores públicos, esse tipo de identificador não é considerado confiável e frequentemente vem acompanhado de um endereço alternativo:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD HTML 3.2//EN"
"http://www.w3.org/TR/HTML/html.dtd">
```

## Declaração de entidade

A declaração de uma entidade permite que se dê um nome a um texto XML. O nome pode ser referido posteriormente para inclusão do texto a ele associado, no ponto de referência.

```
<!ENTITY chap2 SYSTEM "ch02.xml">
```

A referência a uma declaração é feita delimitando o nome da declaração com os caracteres `&` e `;`, como em `&chap2;`.

Quando um documento XML é processado, inicialmente todas as suas declarações externas (as contidas no documento DTD indicado na declaração `DOCTYPE`) são lidas e depois as declarações locais (as contidas entre os colchetes da declaração `DOCTYPE`):

```
<!DOCTYPE doc SYSTEM "/usr/local/xml/dtds/simple.dtd" [
<!ENTITY chap2 SYSTEM "ch02.xml">
] >
```

As entidades do tipo caractere são usadas para designar caracteres difíceis ou impossíveis de serem escritos no texto de um documento XML (por conflitarem com marcações da linguagem por exemplo. Elas são pré-definidas e devem apenas ser referidas por meio do seu código numérico decimal ou hexadecimal. A entidade `&#241;` designa a letra ñ, cujo código decimal é 241; e a entidade `&#xe7;` designa a letra ç cujo código hexadecimal é E7. O ambiente XML dispõe de algumas entidades caractere pré-definidas que podem ser usadas pelo nome, como `&amp;` para o símbolo `&`, `&apos;` para o símbolo `'`, `&gt;` para o símbolo `>`, `&lt;` para o símbolo `<`, e `&quot;` para o símbolo `"`. Há também entidades caracteres nomeadas, como a `&THRON;` que designa o caractere islandês thorn. Essas entidades nomeadas devem ser declaradas interna ou externamente (como a entidade `chap2` no exemplo precedente).

### 2.4.4 Entidades não processadas

Algumas entidades são inseridas no documento, mas não devem ser processadas. Elas são indicadas pelo termo **NDATA** seguido de um qualificador que indica o seu tipo. A palavra-chave **NDATA** informa o processador XML que o conteúdo da entidade é de um tipo diferente do esperado e o qualificador que o segue pode ou não ser usado pela aplicação que processa o documento XML. No exemplo a seguir o qualificador **GIF** indica que a entidade é um arquivo gráfico:

```
<!DOCTYPE doc [  
  <!ENTITY gdlegal SYSTEM "graficos/deplegal.gif" NDATA GIF>  

```

A declaração acima pode ser usada, por exemplo, em locais do documento em que se espera a inserção de um arquivo gráfico:

```
<doc>  
  <para>Grafo de dependências legais</para>  
  <graphic src="&gdlegal;" />  
</doc>
```

### 2.4.5 Seções CDATA

Os caracteres que não podem ser escritos em um documento XML, ou aqueles que por qualquer razão são apenas tediosos de serem repetidos muitas vezes, podem ser especificados em uma entidade **CDATA** (a única expressão que não pode estar em uma seção **CDATA** é o termo delimitador final (]]>)). As seções **CDATA** são usadas em outros elementos, sempre que se deseja que certos caracteres não sejam processados como caracteres de marcação:

```
<para>Then you can say "<![CDATA[if (&x < &y)]]>" and be done with it.  
</para>
```

### 2.4.6 Instruções de processamento

Nos casos em que o documento deve conter informação destinada a certos processadores ou aplicações, essa informação pode ser fornecida por meio de instruções de processamento. Esse tipo de elemento é processado apenas pela aplicação para a qual é destinado, que pode identificá-lo por seu nome (seguinte ao delimitador inicial, <?). Se forem necessários mais argumentos para a aplicação, estes podem ser fornecidos após o nome da entidade:

```
<?prgcalc repetição=9 final?>
```

### 2.4.7 Comentários

Os comentários são escritos entre os termos `<!--` e `-->`.

## 2.5 Espaços de nomeação

Os espaços de nomeação (*namespaces*) definem espaços aos quais uma coleção de identificadores (nomes de atributos e elementos) é associada, permitindo que a referência a eles não se confunda com referências a identificadores de mesmo nome associados a outros espaços de nomeação. Os espaços de nomeação permitem o uso de diferentes terminologias ou vocabulário em um mesmo documento.

Os espaços de nomeação são declarados como atributos de entidades e afetam os elementos desta entidade, incluindo seus descendentes, mas não afetam as entidades ancestrais ou irmãs. A declaração de um espaço de nomeação é feita por meio do atributo `xmlns` seguido de um identificador único (geralmente um endereço no formato URL):

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
```

Uma mesma entidade pode ter vários espaços de nomeação: um padrão e outros identificados por prefixos indicador após o nome:

```
<part-catalog
  xmlns:nw="http://www.nutware.com/"
  xmlns="http://www.bobco.com/"
>
```

Na entidade `<part-catalog>` acima sempre que um elemento pertencer ao primeiro espaço, ele será identificado pelo prefixo `nw`; e quando pertencer ao segundo espaço ele será identificado sem prefixo:

```
<part-catalog xmlns:nw="http://www.nutware.com/" xmlns="http://www.bobco.com/">
  <nw:entry nw:number="1327">
    <nw:description>torque-balancing hexnut</nw:description>
  </nw:entry>
  <part id="555">
    <name>type 4 wingnut</name>
  </part>
</part-catalog>
```

## 2.6 Verificando e validando documentos XML

A verificação sintática de documentos XML pode ser feita por qualquer processador XML, enquanto a validação requer a existência de uma definição que sirva de parâmetro para a verificação da conformidade do documento.

### 2.6.1 Verificação

A verificação de um documento XML testa a sintaxe do documento, indicando a existência de etiquetas mal formadas, caracteres inválidos, elementos incompletos e outros aspectos sintáticos. No ambiente linux, essa verificação pode ser feita pelo programa `xmllint`:

```
%  
xmllint documento.xml
```

### 2.6.2 Validação baseada em tipos de documento

A validação de um documento XML é baseada em uma definição que indica outros aspectos que devem estar presentes no documento, como a ordem e o tipo de cada elemento, a obrigatoriedade e a faixa de valores que cada elemento pode assumir etc. Uma dessas definições é o modelo DTD (do inglês, *Documento Type Definition*). A associação do documento XML ao modelo DTD que especifica as regras de validação é feita por meio de uma declaração DOCTYPE:

```
%  
<!DOCTYPE telegram SYSTEM "dtd/documento.dtd">
```

No ambiente linux, o programa `xmllint` também pode ser usado para testar se um documento XML está em conformidade com sua definição:

```
%  
xmllint --valid documento.xml
```

### 2.6.3 Validação baseada em esquemas XML

A definição de tipos de documentos XML via modelos DTD não permite a validação de alguns aspectos relacionados ao conteúdo, como por exemplo determinar que um elemento obrigatório deve conter informação (e não apenas constar no documento como um elemento vazio). O uso de documentos XSD (do inglês, *XML Schema Definition*) que especificam esquemas de conformidade, permite maior flexibilidade na definição de aspectos de conteúdo, incluindo a definição de padrões textuais que o documento deve exibir.

A validação usando esquemas XML pode ser feita no ambiente linux por meio do programa `xmllint`:

```
xmllint --valid --schema xsd/documento.xsd documento.xml
```

## 2.7 Transformando documentos XML

Os documentos XML podem ser processados de vários modos por programas aplicativos. Para as tarefas mais comuns de apresentação, verificação e validação e transformação em documentos de outros tipos existem aplicativos já disponíveis.

### 2.7.1 Apresentação

O foco da linguagem XML é a descrição do formato e conteúdo do documento. Os aspectos estéticos, de apresentação, são na maioria das vezes tratados externamente por meio de folhas de estilo.

#### Uso de folha de estilo

Uma forma de apresentar o conteúdo de um documento XML é usar uma folha de estilo especificando como cada elemento deve ser posicionado e a aparência que deve ter. A folha de estilo em cascata (CSS, do inglês *Cascading Style Sheet*) pode ser usada por uma aplicação capaz de processar suas orientações.

Para que um documento XML seja associado a uma folha de estilo CSS deve conter uma instrução do tipo `xml-stylesheet`:

```
<?xml-stylesheet type="text/css" href="ex2_memo.css"?>
```

### 2.7.2 Transformando documentos XML

Os documentos XML podem ser transformados, tanto para incorporar elementos de estilo quanto para gerar novos documentos, com diferentes características estruturais.

#### Transformação XLST

Outra forma de incorporar elementos de apresentação a um documento XML é transformá-lo em outro tipo de documento que contenha as informações de estilo desejadas, como um documento HTML, por exemplo. Essa transformação é feita com base em uma especificação XSL (do inglês *eXtensible Stylesheet Language*) que define a correspondência entre os formatos origem e alvo, por um processador XSLT (do inglês *XSL Transformation*).

Existem diversos processadores XLST. Para o ambiente linux, o programa `xsltproc` pode ser usado:

```
xsltproc -o telegrama.html ex02_telegrama.xsl ex02_telegrama.xml
```

A associação de um documento XML à especificação XSL com as definições de estilo a serem usadas na transformação pode ser incorporada ao próprio documento por meio de uma instrução de processamento `xml-stylesheet`. Desse modo, o documento pode ser convenientemente transformado e exibido por aplicações que consigam interpretar esse tipo de instrução, como os navegadores de internet:

```
<?xml-stylesheet type="text/xsl" href="/path/to/stylesheet"?>
```

A transformação baseada em uma folha de estilo XSL é geral, podendo ser usada para transformar o documento XML em outros tipos, não relacionados à simples formatação para apresentação.

### Transformação com Objetos de formatação

A transformação de um documento XML em alguns tipos de documentos cujo formato seja complicado, como documentos PostScript, PDF e RTF, por exemplo, pode ser extremamente trabalhosa se realizada diretamente de uma especificação XSL. Para esses casos é mais conveniente usar uma especificação XSL-FO (do inglês *XSL Formatting Objects*) que define os objetos de formatação — mais adequados à geração de documentos complexos. Um documento XSL-FO é gerado a partir de uma especificação XSL e pode ser usado posteriormente por aplicações para gerar o documento complexo desejado.

Um dos processadores de documentos XSL-FO é o programa `fop`, desenvolvido pela fundação Apache. No exemplo a seguir o documento `ex02_telegrama.xml` é transformado em um documento `pdf`, seguindo a especificação contida em `ex02_telegrama_pdf.xsl`. Nesse exemplo, o documento do tipo XSL-FO, utilizado para produzir o documento PDF, é gerado internamente pelo programa:

```
fop -xml ex02_telegrama.xml -xsl ex02_telegrama_pdf.xsl -pdf telegrama.pdf
```

Por serem do mesmo fabricante e baseados na linguagem java, os programas `fop`, `Xerces`, para verificação e validação de documentos XML, e `Xalan`, para transformações de documentos XSL, são geralmente usados conjuntamente, como parte de um mesmo ambiente de edição e programação.

## 2.8 Programação com documentos XML

Além das ferramentas disponíveis para transformação de documentos XML, podemos desenvolver programas que interajam com tais documentos, extraindo e processando informações para o atendimento de necessidades específicas.

A interação entre programas aplicativos e documentos XML ocorre por meio de processadores gramaticais capazes de percorrer um documento XML, extraindo e fornecendo informações. Há duas estratégias possíveis: a primeira utiliza a técnica de fornecer as informações ao programa a partir do processamento sequencial do documento (técnica *push*). O processador gramatical obtém as informações do documento XML e chama o programa aplicativo para tratar cada elemento obtido. A segunda estratégia utiliza a técnica de obter as informações necessárias a partir do programa aplicativo de modo não sequencial (técnica *pull*). O controle do processamento reside no programa aplicativo que solicita ao processador gramatical a informação desejada.

### 2.8.1 Processamento sequencial

Uma interface de programação (API, do inglês *Application Programming Interface*) que permite o uso da primeira estratégia é a SAX (do inglês, *Simple Api for Xml*). Esta interface permite a criação de controladores para a obtenção de informação de fontes que não sejam XML, como base de dados, permitindo grande flexibilidade no desenvolvimento de aplicativos.

O processamento é sequencial (*stream processing*, simulando o processamento gramatical do documento, não permitindo retorno a posições do documento já processadas.

### 2.8.2 Processamento baseado em árvore

A segunda estratégia pode ser implementada por meio da interface DOM (do inglês, *Document Object Model*) que define vários métodos de acesso aos nodos da árvore que representa a estrutura do documento XML, permitindo que o programa possa percorrer os nodos da árvore, obtendo a informação associada a eles, de modo não sequencial.

### 2.8.3 Outros modos de acesso

Para necessidades mais simples, um programa aplicativo pode interagir com um documento XML usando mecanismos como *XPath*, que permite a localização de informação contida em um determinado ponto do documento XML, ou *XPointer*, que estende o endereçamento do *XPath* permitindo a obtenção de informação em qualquer localização da internet.



## 2.9 Schema XML

Conforme a documentação do W3C [30], um *schema XML* define uma classe de documentos XML, especificando sua estrutura e o conjunto de elementos e atributos válidos para esta classe (ou, seja, a estrutura, conteúdo e semântica da classe, como explanado em [29]). A partir de um schema XML, é possível validar qualquer documento XML, verificando se o mesmo corresponde à classe XML definida pelo schema. Um validador de um schema XML pode ser uma máquina, de modo que é possível construir ferramentas computacionais para automatizar a validação de classes de documentos XML.

É possível estabelecer uma hierarquia de herança utilizando a propriedade de extensibilidade dos schemas XMLs. De forma análoga a uma linguagem de programação orientada a objetos, é possível herdar características de um outro schema XML, enquanto outros elementos mais específicos são definidos.

## 2.10 URI, URL e URN

Nesta seção, são apresentados URI, URL e URN. URIs são um conjunto maior de representações; URLs e URNs são subconjuntos de URIs.

### 2.10.1 URIs

*Uniform Resource Identifier*, ou, em português, *Identificador Uniforme de Recurso*, é uma representação, disposta em uma string de caracteres, usada para identificar um recurso disponível na internet [19]. Um URI possui um sintaxe definida para a resolução do nome, de modo a permitir que uma ferramenta computacional (um *parser*) possa ser capaz de analisar sintaticamente e semanticamente, resolvendo o nome. Uma URI pode ser uma URL ou URN, de forma não excludente, conforme é ilustrado na figura 2.1.

### 2.10.2 URLs

Localizador uniforme de recursos, ou *Uniform Resource Locator* é uma URI que utiliza a localização na internet de um recurso para identificá-lo. É uma solução largamente usada na internet para referenciar documentos, páginas web, arquivos de imagem, vídeos, entre vários recursos. Porém, como a URL se baseia na localização física do recurso referenciado, isto inclui alguns problemas. Entre eles, quando a localização do recurso não estiver mais disponível, a referência será quebrada. Além disso, será impossível fazer referência a documentos ainda não criados., conforme assinalado em [5].

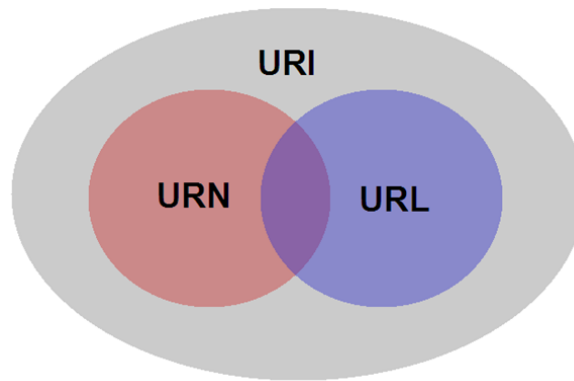


Figura 2.1: Diagrama de Venn para URIs, URNs e URLs [16].

### 2.10.3 URNs

URNs são nomes uniformes de recursos (*Uniform Resource Names*), um subconjunto das URIs. Permite identificar nomes de forma persistente para recursos encontrados na Internet (isto é, o recurso continua acessível através do tempo). Além disso, um URN identifica somente um recurso único bem definido (univocamente).

Referenciar outros documentos normativos é uma necessidade importante. É frequente, por exemplo, a apresentação de leis que fazem referência a outras. Em uma apresentação de um documento normativo em formato eletrônico disponibilizado na internet, é comum e desejável a disponibilidade de consultar imediatamente o documento a que faz referência.

Um URN resolve esses problemas fornecendo um identificador de nome que será resolvido dinamicamente. Um serviço de resolução de nomes realiza o parse, traduzindo a URN para um URL, de forma que uma URL válida será retornada para a aplicação cliente.

# Capítulo 3

## A Importância do Problema

### 3.1 Necessidade de modelar textos legais

O processo de elaboração legislativa e a utilização dos textos legais é de fundamental importância para a própria aplicação da justiça. Com o advento da Internet e o avanço de ferramentas e modelos computacionais, esse processo cedo se apresentou como candidato à automação.

No campo da elaboração de textos, o padrão XML é adotado em larga escala. Há diversas iniciativas para adoção da XML para representar os textos legais. Dada a complexidade inerente a esses textos, algumas dessas iniciativas centram-se em determinados aspectos, deixando outros não tão desenvolvidos. Alguns desses aspectos são citados a seguir:

- **Elaboração:** O processo legislativo de elaboração das leis. Esse é um processo muito dinâmico, centrado em negociação e intenso formalismo.
- **Documentação:** O processo de documentação, descrição da legislação. Este processo depende da legislação de cada país, pois cada um adota uma estrutura legal diferente. No Brasil, a estrutura legal é disciplinada pela Lei Complementar número 95, de 26 de fevereiro de 1998, mas, em certos países, esta estrutura legal é diferente. De certa forma, há, com a internacionalização, uma necessidade de adotar estruturas de representação que possam se adequar às diferentes formas de descrição legal e servir de mapeamento entre elas.
- **Modificação:** O processo de modificação da legislação. Esse aspecto é fundamental, pois as legislações se modificam ao longo do tempo, de forma complexa, O texto legal deve ser capaz de representar essas mudanças.

- Utilização: O processo de utilização das leis também pode ser auxiliado com a estrutura escolhida. Há relação entre normas diversas: normas gerais e específicas para um mesmo tema, bem como normas envolvendo temas correlatos.
- Diferentes categorias: A existência de normas em diferentes categorias: leis, decretos, jurisprudência.
- Acesso: O acesso à legislação, permitindo a todos o conhecimento das normas relacionadas à sua área de interesse (do seu problema). Os projetos que serão apresentados abordam esses aspectos de diferentes modos. Este trabalho discutirá a extensão de cada um.

Este trabalho apresenta algumas das iniciativas de representação de textos legais em XML, discutindo seus pontos fortes e fracos e as semelhanças e diferenças entre eles. Entre os vários aspectos que são abordados pelas propostas, estão:

### **Internacionalização**

O suporte a várias linguagens é um aspecto importante, pois indica se a proposta de padronização é capaz de representar a codificação legal de mais de um país, ou mesmo, se é capaz de representar o mesmo código legal em várias línguas, o que permite ampliar o escopo de acesso da informação. Outras possibilidades são: a capacidade de interoperabilidade entre padrões de representação e utilização do padrão como base para construir outro padrão com propostas mais específicas (extensão do padrão), o que é possível utilizando a capacidade de extensão dos schemas XML. Como exemplo, o schema XML do LexML Brasil estende o schema XML do Akoma Ntoso, o que será posteriormente discutido no capítulo 4. Um aspecto importante para permitir o suporte multilíngue é a codificação usada para representar caracteres. A preocupação com utilização de UTF-8 é presente nas propostas analisadas, conforme será discutido no capítulo 4.

### **Estratégias com relação a alguma proposta anterior**

Como já foi exposto no item anterior, a capacidade de interoperabilidade entre padrões de representação mostra-se como característica desejável entre as propostas analisadas. Cada uma das iniciativas abordadas no capítulo 4 será analisada com relação às escolhas para manter interoperabilidade e manter compatibilidade com alguma proposta anterior.

### **Referenciação**

Os documentos normativos em geral, têm como característica a necessidade de referenciação a outros documentos legais. É necessário, portanto, que uma proposta de represen-

tação se preocupe com esta questão e apresente uma solução que permita ao documento apresentado para visualização, fácil acesso entre todos os documentos referenciados. As propostas analisadas no capítulo 4 adotaram soluções que permitem referenciar um documento de forma unívoca, evitando a formação de referências quebradas.

## 3.2 Propostas para modelagem de textos legais

Segundo a documentação do LexML Brasil [7], o projeto *Norme in Rete* (NIR) foi a primeira iniciativa de padronização de modelagem de textos legais usando XML, com propostas para estruturação e organização de normas jurídicas no formato XML. Foi uma iniciativa do governo italiano com a participação da Universidade de Bolonha, com a implantação tendo início por volta de 2002. A partir do NIR, surgiram outras iniciativas, tais como o CEN Metalex e o Akoma Ntoso, que utilizaram conceitos herdados do Norme In Rete. Como exemplo, Vitali [14] elucida que o modelo do Akoma Ntoso é derivado do modelo do Norme In Rete.

Uma característica marcante entre as várias propostas que surgiram é a tendência a manter schemas XMLs extensíveis e compatíveis com os projetos já iniciados. Desta forma, vislumbra-se uma futura compatibilidade e interoperabilidade entre os padrões. A seguir, são listadas algumas propostas com a finalidade de contextualização:

- Itália - Projeto Norme in Rete (NIR): de acordo com [10], esta iniciativa teve início em 1999 com a Autoridade Italiana Para Tecnologia da Informação na Administração Pública, em conjunto com o Ministério Italiano da Justiça e a Universidade de Bolonha. Define identificadores (URNs) para referência de documentos e DTDs<sup>1</sup> para estruturação dos documentos. Envolve um portal que fornece serviços de busca e disponibilização de documentos legais, bem como a tradução das URNs em URLs.
- África - Akoma Ntoso: Projeto das Nações Unidas no contexto do projeto “Fortalecimento dos Sistemas de Informação dos Parlamentos na África”, de modo a promover o acesso e compartilhamento de código legal e documentos normativos por todos os países participantes, fornecendo, com isto, instrumentos para fortalecimento da democracia na África. Possui grande capacidade de interoperabilidade e é projetado para ser extensível.
- Áustria - eLaw: Consiste de um sistema de workflow que acompanha todo o processo de formação da lei, desde a iniciativa até a publicação em diário oficial eletrônico [7].

---

<sup>1</sup>*Document Type Definition*, assim como um *Schema XML*, é um documento que define a estrutura, os elementos e atributos válidos para um documento XML.

- Dinamarca - LexDania: Utiliza camadas estratificadas que permitem a montagem de múltiplos esquemas, de acordo com uma metodologia que garante a construção de esquemas compatíveis e consistentes [7].
- Europa - Metalex CEN: O Centro Europeu de Normatização (CEN) apresentou esta proposta no formato de workshop em 2007[4].
- Holanda - Metalex: Esquema flexível e extensível para a codificação de normas em XML, influenciou fortemente o Metalex CEN.
- Suíça - CHLexML: Define XML Schema para ser utilizado na publicação de todos os atos legislativos, nos níveis federal, estadual (27 cantões) e local. O sistema lida com o problema de múltiplas línguas oficiais e múltiplos sistemas de classificação (1 para cada cantão) [7].
- Estados Unidos - USLM: *United States Legislative Markup*, ou Linguagem de Marcação Legislativa dos Estados Unidos, é uma iniciativa do congresso americano projetada para representar a legislação dos Estados Unidos. Possui compatibilidade com Akoma Ntoso.
- Brasil - LexML Brasil: projeto de vários órgãos governamentais brasileiros, se propõe a concentrar informação normativa dos três poderes e todas as esferas administrativas no seu portal. Estende (herda) o schema XML do Akoma Ntoso. Possui um sistema para resolução de nomes (URNs) para tratar referênciação.

A seguir, é apresentada a OASIS [23], organização que promove o Akoma Ntoso e incentiva o desenvolvimento de padronização para representação normativa.

## OASIS

A OASIS (*Organization for the Advancing of Open Standards for the Information Society*) é uma organização que trabalha com o desenvolvimento e promoção de padrões para a sociedade da informação, com propostas em vários campos, desde protocolos para troca de mensagens de negócios, aplicações de governo eletrônico, aplicações em realidade aumentada, serviços baseados na web, serviços biométricos, representação de documentos de negócios e muito mais. É considerada a organização líder mundial na padronização relacionada à XML.

LegalXML foi formada em 1998 como resultado da colaboração entre advogados, administradores judiciais e especialistas em tecnologia da informação. Teve como primeiro produto o desenvolvimento de especificações para o preenchimento de demandas judiciais. Em 2002 a iniciativa integrou-se à organização OASIS.

Na área de informática legal há os comitês de marcação de citações legais, documentação legal, regras argumentativas legais e preenchimento eletrônico de demandas judiciais. Todas as iniciativas são coordenadas pela Seção LegalXML.

O comitê técnico LegalDocML tem como propósito promover a adoção de padrões de documentação para documentos legais. O comitê técnico LegalDocML cuida da promoção mundial para o uso do padrão XML nos parlamentos, congressos e cortes de justiça em todo o mundo, catalogando e promovendo as melhores práticas nessa área.

O objetivo do comitê técnico LegalDocML é coletar requisitos da comunidade de usuários e pessoas com interesse (*stakeholders*) na criação, uso e gerência de documentos legais e legislativos, de modo a estender e refinar o padrão de documentação.

O padrão desenvolvido pelo comitê tem sua especificação baseada na experiência da linguagem Akoma Ntoso, mantendo esse nome e adotando como raiz do esquema XML o termo *akomaNtoso*. Um dos propósitos da iniciativa é examinar os relacionamentos entre o vocabulário XML proposto e outros esforços semelhantes, especialmente os que já obtiveram aceitação nacional e aqueles incluídos em outros vocabulários legais XML.

O padrão CEN Metalex oferece um metamodelo conceitual apropriado para o gerenciamento de questões relativas à aderência (e compatibilidade) entre diferentes padrões nacionais XML. O projeto Akoma Ntoso foi concebido para ser conforme ao CEN Metalex.

A política do OASIS é clara ao adotar a conformidade com o padrão CEN Metalex como o primeiro e mais importante requisito para comparação entre a linguagem XML aprovada pelo comitê técnico e outro padrão XML.

Fica claro que este requisito exclui outras iniciativas que não possam ser controladas/incorporadas na linguagem proposta pelo comitê. É um movimento de mercado que será bem sucedido à medida que os membros da organização ganhem espaço no mercado e nos governos.

# Capítulo 4

## Soluções Apresentadas Existentes

### 4.1 CEN Metalex

CEN Metalex, é uma iniciativa de visão padronizada de documentos legais para o propósito de intercâmbio e interoperabilidade de informações no contexto de desenvolvimento de software. O propósito do Metalex é facilitar a troca de informações e a comunicação entre jurisdições com diferentes formatos XML para seus documentos legais. É um padrão que possibilita a interface entre padrões. [20].

Em outras palavras, o CEN Metalex padroniza a maneira na qual documentos legais, legislativos e parlamentares (bem como as referências contidas nestes documentos para outros documentos) são representados em XML. Para isto, está definido um mecanismo para extensão do schema XML disponibilizado, adicionando metadados, referências cruzadas, documentos de componentes incorporados e uma convenção básicas de nomes.

A iniciativa conta com um servidor de documentos que contém vasta documentação legal formatada segundo a especificação <sup>1</sup>. A iniciativa também é adotada em iniciativas de alguns governos: o sítio de legislação do governo britânico incorpora identificação específica para os documentos formatados segundo o padrão CEN MetaLex.

A estrutura do padrão CEN MetaLex contém especificação para modelos de conteúdo, identificadores usados em citações, convenções de nomes, metadados embutidos, esquema de ontologias para web baseado na linguagem OWL [28] e convenções globais de conformidade [12].

O Metalex teve sua origem em um workshop promovido pelo CEN [21]<sup>2</sup>. O acordo

---

<sup>1</sup>Os documentos foram convertidos do portal holandês de legislação e contém o Estatuto de Roma para a Corte Internacional de Justiça, por exemplo.

<sup>2</sup>O CEN é o comitê europeu para padronização. O CEN desenvolve e publica padrões europeus e especificações técnicas que satisfazem as necessidades dos negócios da Europa e de outras organizações. Segundo a página na web do próprio CEN [11], este trabalho traz benefícios, tais como promoção de segurança, qualidade e confiabilidade de produtos, serviços, processos; reforçar o mercado e suportar o crescimento da economia e a difusão de novas tecnologias e inovação.



resultante do workshop, assinado em 2010 ([12]). Porém, este mesmo acordo é claro ao estabelecer que o padrão ainda não foi incorporado como um padrão europeu (parte do CEN). O status atual é de pré-norma para o CEN — uma especificação disponível para o público, objeto de um acordo de trabalho CEN.

## Referenciação

A sintaxe adotada para citações legais é baseada no uso de URI (*Universal Resource Identification*) para todos os tipos de documentos produzidos por parlamentos e cortes e gerenciados por meio do vocabulário XML. Esta sintaxe é a convenção de nomeação adotada e é considerada similar ao identificado de legislação europeu (*European Legislation Identifier*) e consistente com as URI baseadas em http usadas pelo *URN:Lex syntax* (proposto pelo comitê técnico).

## 4.2 Akoma Ntoso

Akoma Ntoso é uma iniciativa do programa da Organização das Nações Unidas/DESA (*Department of Economic and Social Affairs*) para o desenvolvimento de um plano de ação a ser adotado como fundação dos parlamentos baseados na internet da África: *Africa i-Parliament Action Plan*. O projeto Akoma Ntoso é atualmente o principal pacote no contexto do *LegalDocML Technical Committee* no contexto do projeto OASIS.

O *Africa i-Parliaments Action Plan* pretende fortalecer parlamentos africanos e prover suporte para que estes possam aperfeiçoar o cumprimento de suas funções democráticas, e para que se tornem abertos, participativos e baseados em conhecimentos [25].

O nome Akoma Ntoso é um acrônimo para *Architecture for Knowledge-Oriented Management of African Normative Texts using Open Standards and Ontologies*, mas foi escolhido com o duplo sentido de também ser uma expressão da língua Akan da África Ocidental significando corações ligados (*linked hearts*).

Segundo Vitali [14], consiste em um conjunto de diretrizes para serviços informatizados parlamentares em um contexto da África como um todo. Endereça conteúdo da informação e recomenda políticas técnicas e especificações para construir e conectar sistemas de informações parlamentares através da África. O Akoma Ntoso possui um schema XML definido que descreve uma classe de documentos parlamentares (tais como contas, atos e registros parlamentares). A representação do documento por meio desta classe, o que a torna estruturada em elementos e atributos (e o uso do schema XML para validar a classe) permite construir ferramentas computacionais para extrair informações dos documentos normativos representados, ou seja, esses documentos tornam-se "passíveis de leitura por máquinas". Desta forma, é possível promover a interoperabilidade das infor-

mações produzidas pelos parlamentos africanos e o acesso aberto dessa informação para os cidadãos.

O Akoma Ntoso possui três objetivos principais que são necessários para o sucesso do projeto como um todo, como informa Vitali [14]:

- Definir um padrão comum para intercâmbio de dados entre parlamentos;
- Definir as especificações para um modelo de documento no qual sistemas parlamentares podem ser construídos;
- Definir um mecanismo fácil para citações e referências cruzadas de dados entre parlamentos.

E tem como objetivo prover dois tipos básicos de interoperabilidade:

- interoperabilidade semântica tem como enfoque garantir que o significado preciso de informação compartilhada é passível de entendimento por qualquer pessoa ou aplicação recebendo os dados;
- interoperabilidade técnica tem como enfoque garantir que todas as aplicações, sistemas e interfaces relacionados ao Akoma Ntoso são baseadas em um núcleo compartilhado de tecnologias, linguagens e pressupostos técnicos facilitando intercâmbio de dados, acesso de dados e reuso de competências e ferramentas adquiridas [14].

A possibilidade de acessar os documentos legais primários, o material de trabalho parlamentar e os documentos da atividade judicial é colocado como algo que vai além do simples acesso físico às fontes originais. Requer a descrição e classificação da informação de modo uniforme e organizada e estruturada de modo que possa ser utilizada por aplicações de software. Esse é o conceito de *acesso aberto* e permite que aplicações sofisticadas possam ser desenvolvidas, para além da simples exibição na tela do computador [22].

### **Estratégia em relação a propostas anteriores**

O padrão Akoma (como os demais nessa área) pretende ser extensível, de modo a permitir a adoção de adições locais sem que se perca a interoperabilidade com os demais sistemas e usuários. O schema XML do Akoma Ntoso foi explicitamente projetado para ser aderente com o padrão CEN Metalex, um dos padrões de fato. Além disso, ele é derivado do Norme In Rete.

### **Internacionalização**

A padronização pretendida insere-se no contexto da globalização, em que os países devem ser capazes de trocar informações de modo consistente. A consecução desse objetivo pode

levar à imposição de soluções globais sob a justificativa de que as necessidades em certo nível são idênticas ou equiparáveis.

Conforme Vitali [14], o Akoma Ntoso foi criado com a intenção de ser capaz de conectar diversos parlamentos africanos, para o desenvolvimento do projeto da ONU em que foi designado. O benefício de conectar parlamentos por meio de uma representação normativa comum pode impulsionar o desenvolvimento entre eles, à medida que se torna fácil compartilhar experiências bem sucedidas e documentação normativa. Neste contexto, o Akoma Ntoso foi designado com uma natureza multi-linguagem, capaz de suportar ao menos três línguas européias: inglês, francês e português, além de algumas línguas africanas. Todos os documentos em Akoma Ntoso devem ser escritos em codificação UTF-8. Além disso, apesar do schema XML ter sido projetado para ter o inglês como língua padrão, uma lista de nomes equivalentes é fornecida em francês e português. Também é possível fornecer nomes para qualquer outra linguagem. Esta característica favorece o Akoma Ntoso como padrão extensível, pois iniciativas de padronização de línguas diferentes podem estender o schema XML, adaptando suas línguas nativas.

## Referenciação

O Akoma Ntoso possui um sistema de convenção de nomes, chamado de *Akoma Ntoso Naming Convention* [24] e um sistema capaz de resolver os nomes definidos pela convenção, chamado de *Akoma Ntoso IRI Resolver* [3]. Documentação, definições e exemplos podem ser encontrados em seus sítios oficiais.

URIs são usadas pelo Akoma Ntoso para criar referências unívocas e evitar a formação de "links quebrados". A convenção de nomes prevê a possibilidade de referenciar versões diferentes do mesmo documento e inclusive, versões do mesmo documento em línguas diferentes, estendendo o suporte multilíngue para a referenciação, como está definido em [24].

## 4.3 USLM

Nos Estados Unidos a adoção do XML como padrão para a troca de documentos legislativos foi realizada pelo comitê administrativo do Congresso (*Committee on House Administration*) em 1999, logo após um estudo de viabilidade concernente ao uso de XML/SGML como meio de produzir informações legislativas. O propósito é apoiar os documentos considerados legislativos por natureza.

O modelo XML usado pelos Estados Unidos para representar a legislação do Congresso dos Estados Unidos (*United States Legislative Markup*) foi concebido inicialmente para produzir em XML os títulos da codificação americana, mas o padrão pode ser adaptado

para a inserção e produção de apêndices, leis, resoluções e outros materiais legislativos. O projeto da USML usa a linguagem de definição de esquemas XML (XSD, *XML Schema Definition Language*). XHTML é usada para tabelas e outras estruturas de conteúdo, e o *Dublin Core* para a descrição de metadados. SVG é usado para gráficos vetoriais.

O padrão USLM adota dois modelos básicos de documentos: o abstrato e o concreto. O primeiro é geral e flexível e define um conjunto mínimo de *tags*. O segundo é mais específico, derivado do primeiro para definir precisamente o modelo da codificação americana (*United States Code*), usando terminologia bem estabelecida no Escritório do Conselho de Revisão Legal (*Office of the Law Revision Counsel*).

## Aspectos gerais

Em geral, o padrão USLM evita a geração de texto. Baseados em experiências com o uso anterior do padrão XML, os projetistas da linguagem consideram que a geração de texto pode ser problemática, principalmente tratando-se de lei existente (em vigor).

Um aspecto interessante na política adotada pelos Estados Unidos em relação à publicação de sua legislação no formato HTML ou XML, é a aceitação (autorização) de que o código legal em tal formato pode ser usado por terceiros, com a única ressalva que, uma vez fora do controle direto do *Office of the Law Revision Counsel*, nenhuma garantia será dada quanto à acurácia da lei [17].

Outro aspecto prático tratado pela USLM é a ocorrência de estruturas legais anômalas resultante tanto da existência de legislação federal em vigor há séculos (enquanto os estilos de concepção e estruturação evoluíram ao longo do tempo); quanto do fato de que há alguns estatutos federais contendo características que não se conformam a nenhum estilo passado ou presente. O primeiro caso (lei antigas) é tratado com o uso do mecanismo de subclasse do esquema XML, o segundo com o mecanismo de subclasse associado ao uso do atribute `@role` [17].

## Estratégia em relação ao padrão Akoma Ntoso

A estratégia adotada foi projetar USLM para ser consistente com o padrão Akoma Ntoso tanto quanto possível, sem entretanto tornar a linguagem um derivativo ou subconjunto do padrão Akoma. Muitos dos nomes de elementos e atributos no USLM casam com os nomes equivalentes do Akoma Ntoso. A ideia é apenas fazer com que seja possível a geração da codificação americana, caso o padrão Akoma torne-se padrão de fato e haja demanda para isso. Para que isto seja possível, o schema XML do USLM estende o schema XML do Akoma Ntoso.

## **Internacionalização**

Como o schema XML da USLM estende o schema XML do Akoma Ntoso, USLM herda suas características multilinguagem. A documentação do schema XML do USLM [17] afirma que o USLM foi criado para representar a legislação do congresso dos Estados Unidos, porém, deve ser capaz também de referenciar outros documentos legislativos que existam em outros formatos XML. Esta característica pode justificar a necessidade de internacionalização herdada do schema XML do Akoma Ntoso.

## **4.4 LexML Brasil**

Segundo a documentação oficial do LexML Brasil [6], é uma iniciativa de diversos órgãos governamentais brasileiros, liderados pelo Senado Federal, com o objetivo de identificar e estruturar as informações legislativas e jurídicas através da integração de processos de trabalho e compartilhamento de dados utilizando padrões abertos, nas três esferas administrativas (federal, estadual e municipal) e entre os órgãos dos três poderes da República (Executivo, Judiciário e Legislativo).

O LexML Brasil possui um portal que reúne diversos documentos normativos (leis, decretos, acórdãos, súmulas, projetos de leis, entre outros documentos) de forma a centralizar a informação disponibilizada em outros portais na Internet de órgãos governamentais brasileiros. A documentação oficial [6] cita ainda que o projeto tem intenção de ser uma rede de informação legislativa e jurídica que pretende organizar, integrar e dar acesso às informações disponibilizadas nos diversos portais de órgãos do governo na Internet.

## **Governo Eletrônico**

O governo eletrônico [15] é um programa do governo federal brasileiro, iniciado a partir do ano 2000, que se utiliza de tecnologias da informação e comunicação para ampliar o acesso do cidadão à informação. No sítio oficial do programa são citadas três idéias fundamentais que orientam os seus objetivos: participação cidadã; melhoria do gerenciamento interno do estado e integração com parceiros e fornecedores. O sítio também enumera a preocupação com a transparência e qualidade dos serviços e informações públicas prestadas e se coloca como instrumento da sociedade para se informar e se manifestar frente às ações governamentais, proporcionando, portanto, uma administração pública mais eficiente. O LexML se insere neste contexto, pois o governo eletrônico confere a ele, atualmente, o estado de "adotado", o que significa que o uso do LexML será incentivado dentro do conjunto de padrões que integra a estratégia do governo eletrônico.

## Aspectos Gerais

A identificação da informação permite, entre outras coisas, o estabelecimento de links entre documentos. No LexML, cada documento legislativo e jurídico possui um identificador unívoco e persistente (chamado URN), que pode ser referenciado sem o temor de que o endereço seja alterado no futuro, gerando "link quebrado" (erro HTTP 404). A identificação permite ainda agrupar as diversas versões de um texto disponíveis nos diversos órgãos, retornando para o cidadão apenas uma página referente a cada documento. Por exemplo, ao pesquisar "Código de Defesa do Consumidor", o Portal LexML retornará uma única página com links para as ocorrências deste documento nos sítios de órgãos do governo, tais como Imprensa Nacional, Senado Federal, Câmara dos Deputados e Presidência da República [6].

A fim de estabelecer referências de hipertextos e remissivas, são utilizados os URNs, entendidos como nomes uniformes atribuídos para cada documento de forma que cada documento tenha o seu identificador único.

Normas jurídicas tais como leis, decretos e instruções normativas, não são documentos isolados uns dos outros. Nosso ordenamento jurídico é composto pelo conjunto de normas emanadas por vários órgãos e publicadas em periódico oficial. Alguns elementos permitem caracterizar o conjunto de normas como um sistema coeso [6]:

- Citações: é comum, dentro do próprio texto normativo, ocorrerem citações para outras normas. Por exemplo, a Lei Federal nº 11.705 realiza remissão para a Lei Federal nº 9.294. Citações são facilmente convertidas em hiperlinks, de forma que o conjunto de normas "antes independentes" se torna um hipertexto, um banco de dados onde as normas podem relacionar-se entre si.
- Dependências diretas: algumas normas citam outras para revogá-las ou detalhá-las. O conjunto de normas relacionadas formam uma sistema coeso. Essas dependências também são facilmente registradas em banco de dados, e dão origem ao chamado "texto compilado" da norma original.
- Dependências hierárquicas: usualmente normas mais específicas e detalhadas regulamentam as normas mais gerais. Essa é uma relação típica entre, por exemplo, as Leis Federais e a Constituição da República. A hierarquia, num banco de dados, permite organizar de forma mais coerente grandes quantidades de normas.
- Vinculação por assunto: apesar de ser um dos aspectos mais difíceis de se estabelecer pela via da automação, o agrupamento de normas em função do assunto que tratam é de grande importância. Metodologias da Biblioteconomia e da Jurisprudência

garantem a confiabilidade de procedimentos computacionais assistidos por pessoas especializadas.

Boa parte dessas informações se encontram explicitadas no texto da norma. O Projeto LexML estabeleceu tecnologias e recomendações para o registro de citações que permitem a criação de hiperlinks persistentes, o estabelecimento de relacionamentos semânticos entre documentos e a semi-automação da geração de texto compilado [6].

O LexML provê tanto a garantia de sempre estar lá, pois ele foi criado para esse fim, como mecanismos padronizados para referenciar não apenas endereços, mas nomes específicos de documentos [6].

### **Estratégia em relação ao padrão Akoma Ntoso**

O schema XML do LexML Brasil estende o schema XML do Akoma Ntoso. Desta maneira, sua estrutura é preservada, ao mesmo tempo que é possível construir elementos específicos para tratar com a documentação normativa brasileira. A documentação oficial [9] informa que o Akoma Ntoso é uma experiência bem sucedida para padronizar documentos normativos de vários parlamentos africanos.

### **Internacionalização**

O suporte multilígue é herdado do Akoma Ntoso. Além disso, a documentação do LexML Brasil [9] afirma que o UTF-8 é adotado para possibilitar armazenamento de documentos textuais de outros países que utilizam conjuntos de caracteres não latinos. Ou seja, o suporte multilígue é explicitamente adotado pelo LexML Brasil.

### **Referenciação**

O LexML Brasil adota a solução de URNs, que serão traduzidas por um serviço de resolução de nomes, retornando URLs com a localização dos documentos desejados. Desta forma, este serviço garante que somente uma URL válida será retornada. Na documentação do LexML Brasil, encontra-se um documento que define a identificação de normas, julgados e projetos de normas através de um nome uniforme [8]. Ao fornecer ao portal LexML Brasil uma URN, este nome será traduzido, retornando o link para o documento desejado (URL). Na figura 4.1 é mostrado um exemplo de URN utilizada pelo LexML Brasil e cada um de seus elementos são destacados.

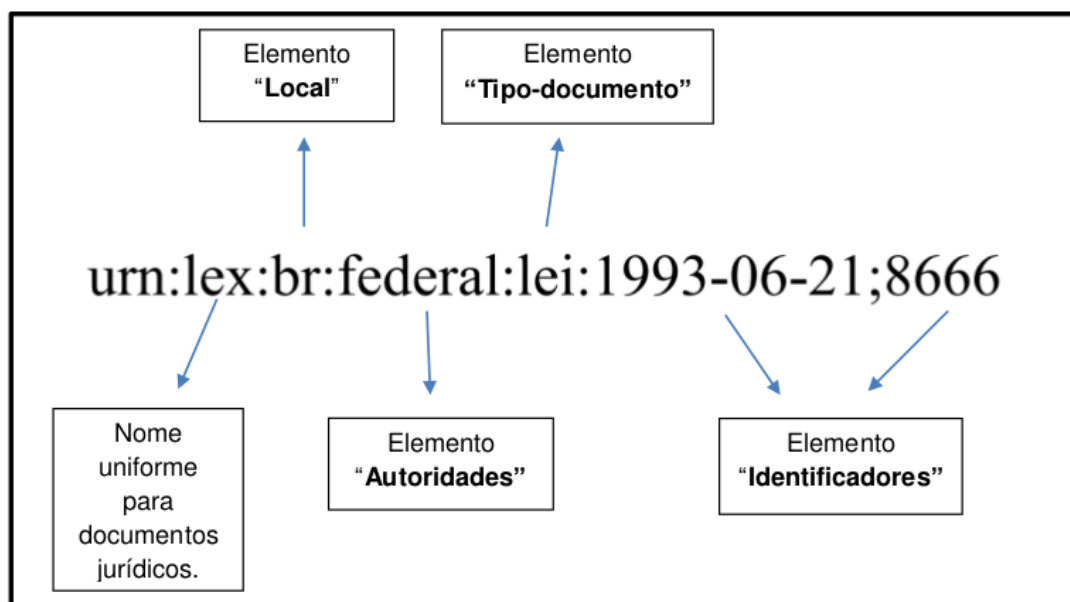


Figura 4.1: Exemplo de URN do LexML Brasil [13].



# Capítulo 5

## Análise Crítica

### 5.1 O que foi proposto

Este trabalho apresenta um levantamento das propostas de padronização de documentos normativos, legislativos e parlamentares no Brasil, na África (no contexto da Organização das Nações Unidas), dos Estados Unidos e da Europa, as compara, a fim de entender as características em comum, os problemas enfrentados por cada uma delas, as soluções adotadas, os motivos de adoção destas soluções e as divergências entre essas propostas. Com isto, faz uma análise do panorama atual para entender as perspectivas e motivações da área para o futuro.

### 5.2 O que foi obtido

Foram listados, em linhas gerais, alguns projetos de padronização existentes, como o *Norme in Rete*, CEN Metalex, Akoma Ntoso, USLM e LexML Brasil, entre outros.

Algumas das propostas foram analisadas, a saber, CEN Metalex, Akoma Ntoso, a USLM (iniciativa do congresso americano) e LexML Brasil. Também foram brevemente apresentadas as organizações OASIS e o comitê CEN, responsáveis pelo Akoma Ntoso e CEN Metalex, respectivamente. Para cada uma das propostas enumeradas, foram apresentadas as características gerais, a estratégia em relação ao padrão Akoma Ntoso, a estratégia de internacionalização e o modelo de referência para outros documentos.

### 5.3 Pontos semelhantes entre as propostas

Verificou-se o cuidado com a internacionalização e o suporte para utilização de diferentes línguas para todas as propostas, inclusive com adoção do *UTF-8* como representação de caracteres.

Também foi possível observar que todas as propostas tiveram o cuidado de manter compatibilidade com alguma proposta anterior. Como exemplificação, o *LexML Brasil* estende o *schema XML* do *Akoma Ntoso*. Esta característica de extensibilidade dos *schemas XML*, explorada por todas as propostas lidas, permitiu que as propostas sejam compatíveis e compartilhem características em comum, enquanto possuem elementos específicos para tratar questões relacionadas ao modelo e aos problemas a que se propõem resolver, referentes aos países e regiões para o qual foram criadas. Isto permite um alto grau de interoperabilidade.

Evidencia-se, desta maneira, que todas estas propostas convergem para um formato onde um modelo de padronização ocupa o lugar de "modelo universal", ou mais geral, a partir do qual modelos regionais serão criados, estendendo o modelo dito universal, conservando a estrutura deste enquanto cria elementos mais específicos. Por apresentar especial cuidado no projeto, e devido às características de ser uma proposta que abrangesse quase todo o continente africano, e além disso, ser uma iniciativa da Organização das Nações Unidas, o *Akoma Ntoso* parece ser a proposta ideal para ocupar esta posição de "modelo universal". De fato, todas as propostas apresentadas neste trabalho que foram concebidas em data posterior ao *Akoma Ntoso*, o adotam como modelo geral e seus *schemas XML* estendem o *schema XML* do *Akoma Ntoso*.

Também foi verificado que todas as propostas apresentadas tiveram o cuidado de estabelecer um mecanismo de referência entre documentos. A preocupação com a referência mostrou-se um aspecto central entre todas as propostas envolvidas, evidenciando assim, a importância que a referência entre documentos exerce no contexto da padronização e representação de documentos legais. Não só o mecanismo escolhido para referência foi motivo de especial cuidado e preocupação para os projetos, mas a infraestrutura usada para manter o mecanismo de referência também foi projetado, em todas as propostas.

Todas as propostas se preocuparam com a possibilidade de referências "quebradas", ou seja, referências que apontam para um recurso inexistente, seja porque este recurso teve sua localização física alterada, seja porque este recurso ainda não foi criado (considerando a possibilidade de referenciar um recurso ainda não finalizado e disponível). A solução deste problema convergiu para adoção das tecnologias *Uniform Resource Identifier* (URI) e *Uniform Resource Name* (URN) para a referência. Também foi necessário criar convenções e sintaxe para o nome. Para esta solução, as propostas criaram infraestruturas capazes de realizar o parse (análise léxica e sintática), resolver e identificar o recurso apontado e encontrar o recurso referenciado de forma unívoca para o nome. Como exemplo, o LexML Brasil criou uma estrutura denominada *Serviço de Resolução de URN*.

## 5.4 Pontos ou abordagens divergentes

As diferenças entre as propostas, basicamente, se resumem nas escolhas específicas projetadas para solucionar questões regionais inerentes à localidade de origem de cada proposta. São escolhas que permitem satisfazer as diferenças encontradas nas leis de cada país ou localização para os quais a proposta foi originalmente criada.

Além disso, cada proposta adotou um mecanismo de convenção e resolução de nomes diferente (embora, fundamentalmente, todas usem URIs e URNs). Basicamente, cada uma criou uma infraestrutura diferente para prover essa solução, ao invés de uma solução única que abrangesse todos os modelos. É possível inferir que, além de diferenças entre os documentos abordados por cada proposta, que tornariam mais fácil e menos oneroso construir uma infraestrutura específica ao invés de uma geral, questões governamentais tornam a propriedade e a posse da infraestrutura de resolução de nomes importante, levando os projetistas a adotarem essa perspectiva.

Como estas propostas optaram por manter interoperabilidade e usar a extensão de *schemas XML*, as demais divergências entre estes modelos são mínimas.

# Capítulo 6

## Conclusão

### 6.1 Revisão do panorama atual

É possível observar, através de notícias de jornais e meios de comunicação em geral, que a globalização e as relações entre países têm favorecido e exigido o desenvolvimento do direito internacional e a necessidade de representação de documentos normativos por meio eletrônico, para a apresentação de forma ampla e de acesso facilitado para pessoas não só do país de origem do documento, como também de outros países. Com isto, e com os resultados obtidos neste trabalho, conclui-se que existe uma tendência global para a padronização de documentos legais usando XML. Várias alternativas para padronização surgiram ao longo dos últimos anos. A própria Organização das Nações Unidas é responsável por uma proposta de padronização, a *Akoma Ntoso*, analisada por este trabalho.

Também é possível concluir com este trabalho, em especial com a análise do capítulo 5, que as diferentes iniciativas pretendem ser interoperáveis entre si. Todas as iniciativas analisadas no capítulo 4 visam a utilização de suporte multilíngue e *UTF-8* para permitir a internacionalização.

Também é conclusivo que todos os padrões pretendem ser compatíveis com os padrões cujos processos de adoção já estão adiantados. Um exemplo é o padrão *LexML Brasil*, que é compatível com o *Akoma Ntoso*, que por sua vez, é compatível com o *CEN Metalex*. A extensibilidade dos *schemas XMLs* permitiu essa cadeia de herança de compatibilidades.

Outra conclusão pode ser obtida da análise do texto do parágrafo anterior e do capítulo 5: existe uma tendência a adotar um destes modelos como "modelo universal", a partir do qual todos os modelos criados para atender problemas de localidades específicas se estendem, preservando a estrutura do modelo mais geral e criando elementos que atendam às demandas específicas. Este cenário favorece a criação de ferramentas que possam atender todas as representações de forma geral, assim como padrões visuais de apresentação dos documentos. No momento em que este trabalho foi escrito, o *Akoma*

*Ntoso* está se consolidando como o "modelo universal" escolhido, pelos motivos também elucidados no capítulo 5.

Outra conclusão deste trabalho é que a referenciação de outros documentos é uma das questões mais importantes tratadas pelos modelos de representação de documentos legais, na medida que documentos legais, por sua estrutura e natureza têm a referenciação como construção chave. Além disso, a solução de referenciação escolhida influencia na possibilidade ou não da obtenção de referências quebradas, possibilidade de versionamento dos documentos referenciados (isto é, possibilidade de referenciar uma ou mais versões específicas de um documento legal, dado que documentos legais constantemente têm seus textos alterados). Este contexto foi analisado no capítulo 5.

## 6.2 Propostas de trabalhos futuros

A partir das informações fornecidas e reunidas neste trabalho, é possível apresentar algumas sugestões de trabalhos futuros para a continuação da pesquisa e estudos no tema abordado:

- Estudo de caso: investigar como as representações (ou as propostas adotadas) atingem os objetivos de cada iniciativa. Por exemplo, no caso do LexML Brasil, pode-se verificar se o acesso à informação é igualmente útil tanto para o público em geral, quanto para os representantes da área de direito;
- Estudar de forma aprofundada os modelos de referenciação de documentos e as soluções (se estas existirem) que as propostas já criadas possuem para versionamento de documento. É um problema conhecido do direito a mudança de textos legais com o tempo. Mudanças em leis e documentos normativos são esperadas, e uma proposta de representação destes documentos deve tomar especial cuidado com relação à referenciação de versões diferentes do mesmo documento. Este mecanismo também deve se refletir na escolha da solução e infraestrutura de identificação e resolução de nomes adotada, para que seja possível referenciar uma versão específica do documento.
- Estudar como as propostas contribuem para as metodologias de dados abertos governamentais. Iniciativas de dados abertos permitem melhorias da transparência e qualidade de serviços governamentais prestados, pois permitem a qualquer cidadão acessar dados públicos, bem como utilizar ferramentas de software para extrair informações.

# Referências

- [1] Constituição de 1988 Publicação Original Portal Câmara dos Deputados. <http://www2.camara.leg.br/legin/fed/consti/1988/constituicao-1988-5-outubro-1988-322142-publicacaooriginal-1-pl.html>. acesso em junho de 2017. 1
- [2] Constituição Portal do Palácio do Planalto. [http://www.planalto.gov.br/ccivil\\_03/Constituicao/Constituicao.htm](http://www.planalto.gov.br/ccivil_03/Constituicao/Constituicao.htm). acesso em junho de 2017. 1
- [3] Akoma Ntoso. Akoma Ntoso IRI Resolver. <http://akresolver.cs.unibo.it/>, 2016. acesso em julho de 2017. 27
- [4] Fabio Vitali Monica Palmirani Alexander Boer, Erik Hupkes e Balázs Rátaí. Cen metalex workshop proposal. 2003. 22
- [5] T. Berners-Lee, R. Fielding, e L. Masinter. RFC 3986, Uniform Resource Identifier (URI): Generic Syntax. <https://tools.ietf.org/html/rfc3986>, 2005. acesso em julho de 2017. 17
- [6] LexML Brasil. Projeto LexML - Documentação. <http://projeto.lexml.gov.br/>. acesso em junho de 2017. 29, 30, 31
- [7] LexML Brasil. LexML Brasil: Apresentação. <http://projeto.lexml.gov.br/documentacao/Apresentacao.pdf>, 2008. acesso em junho de 2017. 21, 22
- [8] LexML Brasil. LexML Brasil: Parte 2 – LexML URN. <http://projeto.lexml.gov.br/documentacao/Parte-2-LexML-URN.pdf>, 2008. acesso em junho de 2017. 31
- [9] LexML Brasil. LexML Brasil: Parte 3 – LexML XML Schema. <http://projeto.lexml.gov.br/documentacao/Parte-3-XML-Schema.pdf>, 2008. acesso em julho de 2017. 31
- [10] P. Spinoso M. Taddei C. Biagioli, E. Francesconi. The NIR project: Standards and tools for legislative drafting and legal document web publication. 2003. ITTIG – Istituto di Teoria e Tecniche per l’Informazione Giuridica C.N.R. Firenze, Italia. 21
- [11] CEN. CEN: Fields of work. <https://www.cen.eu/work/areas/pages/default.aspx>. acesso em junho de 2017. 24

- [12] CEN. Metalex (open XML interchange format for legal and legislative resources). CEN workshop agreement, fevereiro 2010. <ftp://ftp.cen.eu/CEN/Sectors/List/ICT/CWAs/CWA15710-2010-Metalex2.pdf>. 24, 25
- [13] Sâmara Roberta de Sousa Castro. A organização da informação jurídica em meio eletrônico sob o aspecto da Representação da Informação: um estudo de caso sobre o Projeto LexML Brasil. Dissertação (Mestrado), Universidade de Brasília, 2017. vii, 32
- [14] Fabio Vitali e Flavio Zeni. Towards a country-independent data format: the Akoma Ntoso experience. 2007. 21, 25, 26, 27
- [15] Governo Eletrônico. Governo eletrônico - histórico. <https://www.governoeletronico.gov.br/sobre-o-programa/historico>. acesso em junho de 2017. 29
- [16] David P. Heitmeyer. The Internet and the World Wide Web, Names and Locations: URLs, URIs, and URNs. [http://cscie12.dce.harvard.edu/lecture\\_notes/2009/20090128/images/uriurnurl.png](http://cscie12.dce.harvard.edu/lecture_notes/2009/20090128/images/uriurnurl.png). acesso em julho de 2017. vii, 18
- [17] US House of Representatives. *United States Legislative Markup: user guide for the USLM schema*. US House of Representatives, Office of the Law Revision Council, 0.1.4 edition, outubro 2013. 28, 29
- [18] ISO. Information processing – text and office systems – standard generalized markup language (sgml). ISO 8879, International Organization for Standardization, Geneva, Switzerland, 1986. [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=16387](http://www.iso.org/iso/catalogue_detail.htm?csnumber=16387). 3
- [19] Michael H. Mealling e Ray Denenberg. Report from the Joint W3C/IETF URI Planning Interest Group: Uniform Resource Identifiers (URIs), URLs, and Uniform Resource Names (URNs): Clarifications and Recommendations. <https://rfc-editor.org/rfc/rfc3305.txt>, 2002. acesso em julho de 2017. 17
- [20] CEN Metalex. Cen metalex: Open xml interchange format for legal and legislative resources. <http://www.metalex.eu/>. acesso em junho de 2017. 24
- [21] CEN Metalex. Cen metalex workshop agreement. <ftp://ftp.cen.eu/CEN/Sectors/List/ICT/CWAs/CWA15710-2010-Metalex2.pdf>. acesso em junho de 2017. 24
- [22] Akoma Ntoso. Akoma Ntoso: What it is. <http://www.akomantoso.org/akoma-ntoso-in-detail/what-is-it/>, acesso em junho de 2017. 2, 26
- [23] OASIS. Oasis - advancing open standards for the information society. <https://www.oasis-open.org/>. acesso em junho de 2017. 22
- [24] OASIS. Akoma Ntoso Naming Convention Version 1.0. <http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/akn-nc-v1.0.html>, 2017. acesso em julho de 2017. 27

- [25] ONU. Africa i-parliaments action plan: open, participatory and knowledge based parliaments. <https://publicadministration.un.org/parliaments/>. acesso em junho de 2017. 25
- [26] Erik T. Ray. *Learning XML*. O'Reilly & Associates, Inc., Sebastopol, CA, USA, 2 edition, 2003. 4, 5, 6, 7, 8
- [27] W3C. Extensible markup language (xml) 1.0 (fifth edition). <https://www.w3.org/TR/xml/>. acesso em junho de 2017. 3, 4
- [28] W3C. Owl web ontology language overview. <https://www.w3.org/TR/owl-features/>. acesso em junho de 2017. 24
- [29] W3C. W3C: XML Schema. <https://www.w3.org/XML/Schema>. acesso em julho de 2017. 17
- [30] W3C. W3C: XML Schema. <https://www.w3.org/TR/2004/REC-xmlschema-0-20041028/>, 2014. acesso em julho de 2017. 17



# Apêndice A

## Exemplo - Lei de São Salvador

O documento a seguir é um exemplo de lei de El Salvador disponível no sítio do Akoma Ntoso.

### A.1 Ley 604 de 1993 da República de El Salvador

```
<?xml version="1.0" encoding="UTF-8"?>
<akomaNtoso xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://docs.oasis-open.org/legaldocml/ns/akn/3.0/
  CSD13 ./akomantoso30.xsd" xmlns="http://docs.oasis-open.org/legaldocml/
  ns/akn/3.0/CSD13">
  <!-- Task funded by UN/DESA. Analysis and markup performed by Monica
  Palmirani, CIRSFD, University of Bologna -->
  <!-- Paragraph before the main structure -->
  <act contains="originalVersion" name="publicLaw">
    <meta>
      <identification source="#somebody">
        <FRBRWork>
          <FRBRthis value="/sv/decreto/1993-07-15/604/main"/>
          <FRBRuri value="/sv/decreto/1993-07-15/604"/>
          <FRBRdate date="1993-07-15" name="Enactment"/>
          <FRBRauthor href="#presidente" as="#author"/>
          <FRBRcountry value="sv"/>
        </FRBRWork>
        <FRBRExpression>
          <FRBRthis value="/sv/decreto/1993-07-15/604/esp@/main"/>
          <FRBRuri value="/sv/decreto/1993-07-15/604/esp@"/>
          <FRBRdate date="2011-01-10" name="Expression"/>
          <FRBRauthor href="#somebody" as="#editor"/>
          <FRBRlanguage language="esp"/>
        </FRBRExpression>
```

```

<FRBRManifestation>
  <FRBRthis value="/sv/decreto/1993-07-15/604/esp@/main.xml"/>
  <FRBRuri value="/sv/decreto/1993-07-15/604/esp@/main.akn"/>
  <FRBRdate date="2011-01-11" name="XMLConversion"/>
  <FRBRauthor href="#somebody" as="#editor"/>
</FRBRManifestation>
</identification>
<publication date="1993-08-16" name="Diario Oficial número 8, tomo
370" showAs="Diario Oficial"/>
<lifecycle source="#somebody">
  <eventRef eId="e1" date="1993-10-15" source="#ro1" type="generation
"/>
</lifecycle>
<references source="#somebody">
  <original eId="ro1" href="/sv/ley/1993-07-15/604/esp@" showAs="
Original"/>
  <TLCRole eId="presidente" href="/ontology/organizations/sv/
presidente" showAs="Presidente de la República"/>
  <TLCRole eId="pv" href="/ontology/organizations/sv/
primerVicepresidente" showAs="Primer Vicepresidente"/>
  <TLCRole eId="tv" href="/ontology/organizations/sv/
tercerVicepresidente" showAs="Tercer Vicepresidente"/>
  <TLCRole eId="author" href="/ontology/roles/sv/author" showAs="
Author of Document"/>
  <TLCRole eId="editor" href="/ontology/roles/ita/editor" showAs="
Editor of Document"/>
  <TLCPerson eId="Zepeda" href="/ontology/persons/sv/editors/zepeda"
showAs="CIRO CRUZ ZEPEDA PEÑA"/>
  <TLCPerson eId="Herniquez" href="/ontology/persons/sv/authority/
Herniquez" showAs="JOSÉ MANUEL MELGAR HENRÍQUEZ"/>
  <TLCPerson eId="Lopez" href="/ontology/persons/sv/authority/cervone
" showAs="JOSÉ FRANCISCO MERINO LÓPEZ"/>
  <TLCPerson eId="somebody" href="/ontology/persons/ita/editors/
somebody" showAs="somebody"/>
</references>
</meta>
<preface>
  <p>
    <docTitle>
      LEY DE FOMENTO Y PROTECCION DE LA PROPIEDAD INTELECTUAL
    </docTitle>
    <docType>Decreto Legislativo</docType>
    <docNumber>No. 912</docNumber>, <docDate date="2005-12-14">del 14
de diciembre de 2005<eol/>
    </docDate>

```

<docProponent>LA ASAMBLEA LEGISLATIVA DE LA REPUBLICA DE EL  
SALVADOR,<eol/>  
</docProponent>

Decreto número 604, del 15 de julio de 1993. <eol/>  
Ley de Fomento y Protección de la Propiedad Intelectual de  
1993 <eol/>

(Publicado en el Diario Oficial número 150 tomo 320 del 16  
de agosto de 1993)<eol/>

Reformado por medio del Decreto Legislativo número 912 de  
14 de diciembre de 2005 <eol/>

(Publicado en el Diario Oficial número 8, tomo 370 del 12  
de enero de 2006, <eol/>

cambiando su nombre a Ley de Propiedad Intelectual.)<eol/>

</p>

</preface>

<preamble>

<container eId="ctn1" name="preamble">

<blockList eId="lst1">

<listIntroduction>CONSIDERANDO</listIntroduction>

<item eId="lst1-itm1">

<num>I.</num>

<p>Que el inciso segundo del Art. 103 de la Constitución,  
reconoce la propiedad intelectual y artística, por el tiempo y en la  
forma determinados por la Ley;

</p>

</item>

<item eId="lst1-itm2">

<num>II.</num>

<p>Que el inciso tercero del Art. 110 de la Constitución,  
establece que se podrá otorgar privilegios por tiempo limitado a los  
descubridores e inventores y perfeccionadores de los procesos  
productivos;</p>

</item>

<item eId="lst1-itm3">

<num>III.</num>

<p>Que en vista del desarrollo alcanzado por tales materias, es  
necesario dictar nuevas disposiciones legales que protejan y regulen  
aspectos de suma importancia como lo son entre otros, la gestión  
colectiva, la protección de los modelos de utilidad, diseños  
industriales, secretos industriales y comerciales, que la legislación  
vigente no comprende; </p>

</item>

<item eId="lst1-itm4">

<num>IV.</num>

```

        <p>Que tanto la Propiedad Literaria , Artística o Científica ,
como la Propiedad Industrial , son las dos ramas que forman la Propiedad
Intelectual , por lo que todas las disposiciones que regulan tales
materias pueden reunirse en un solo cuerpo legal;
</p>
    </item>
</blockList>
<p class="centrado">POR TANTO,</p>
    <p>en uso de sus facultades constitucionales y a iniciativa del
Presidente de la República , por medio de los Ministros de Economía y de
Justicia y de los Diputados Raul Manuel Somoza Alfaro , Gerardo Antonio
Suvillaga , Santiago Vicente Di-Majo, Miriam Eleana Dolores Mixco Reyna,
Jorge Alberto Carranza , Rafael Antonio Morán Orellana y Marcos Alfredo
Valladares Melgar, </p>
    <p class="centrado">DECRETA las siguientes:</p>
</container>
</preamble>
<body>
    <paragraph eId="prg1">
        <content>
            <p>
                <docTitle>LEY DE FOMENTO Y PROTECCION DE LA PROPIEDAD
INTELECTUAL
                </docTitle>
            </p>
        </content>
    </paragraph>
    <title eId="tlt1">
        <num>TITULO PRIMERO </num>
        <chapter eId="chp1">
            <num>CAPITULO UNICO </num>
            <heading>DISPOSICIONES PRELIMINARES</heading>
            <article eId="art1">
                <num>Art.1</num>
                <clause eId="art1-cls1">
                    <content>
                        <p>
                            Las disposiciones contenidas en la presente ley tienen por
objeto asegurar una protección suficiente y efectiva de la propiedad
intelectual , estableciendo las bases que la promuevan, fomenten y
protejan.
                        </p>
                    </content>
                </clause>
                <clause eId="art1-cls2">

```

```

        <content>
        <p>
        La Propiedad Intelectual comprende la propiedad literaria , artí
stica , científica e industrial.

```

```

        </p>
        </content>
    </clause>
</article>
<article eId="art2">
    <num>Art. 2.</num>
    <clause eId="art2-cls1">
        <content>
        <p>
        En caso de conflicto , tendrán aplicación preferente sobre
las disposiciones de esta Ley, las contenidas en los tratados y
convenios internacionales ratificados por El Salvador. </p>

```

```

        </content>
    </clause>
</article>
<article eId="art3">
    <num>Art. 3.</num>
    <clause eId="art3-cls1">
        <content>
        <p>

```

La presente ley no se aplicará a las marcas, nombres comerciales y expresiones o señales de propaganda, las cuales se rigen por el Convenio Centroamericano para la Protección de la Propiedad Industrial, del cual El Salvador es parte.

```

        </p>
        </content>
    </clause>
</article>
</chapter>
</title>
<title eId="tlt2">
    <num>TITULO SEGUNDO</num>
    <heading>PROPIEDAD ARTISTICA, LITERARIA O CIENTIFICA</heading>
    <chapter eId="chp2">
        <num>CAPITULO I</num>
        <heading> – NATURALEZA Y SUJETOS</heading>
        <article eId="art4">
            <num>Art. 4.</num>
            <clause eId="art4-cls1">

```

<content>

<p>

El autor de una obra literaria , artística o científica , tiene sobre ella un derecho de propiedad exclusivo , que se llama derecho de autor .

</p>

</content>

</clause>

</article>

<article eId="art5">

<num>Art. 5.</num>

<clause eId="art5-cls1">

<content>

<p>

El derecho de autor comprende facultades de orden abstracto , intelectual y moral que constituyen el derecho moral; y facultades de orden patrimonial que constituyen el derecho pecuniario.

</p>

</content>

</clause>

</article>

<article eId="art6">

<num>Art. 6.</num>

<clause eId="art6-cls1">

<content>

<p>

El derecho moral del autor es imprescriptible e inalienable y comprende las siguientes facultades:</p>

<blockList eId="art6-cls1-lst1">

<item eId="art6-cls1-lst1-itma">

<num>a)</num>

<p>La de publicar su obra en la forma , medida y manera que crea conveniente; </p>

</item>

<item eId="art6-cls1-lst1-itmb">

<num>b)</num>

<p>La de ocultar su nombre o usar seudónimo en sus publicaciones; </p>

</item>

<item eId="art6-cls1-lst1-itmc">

<num>c)</num>

<p>La de destruir , rehacer , retener o mantener inédita la obra; </p>

</item>

```

        <item eId="art6-cla1-lst1-itmd">
            <num>d</num>
            <p>La de retractarse , o sea de recuperar la obra ,
modificarla o corregirla después de que haya sido divulgada , pero esta
facultad no podrá ejercerla sin indemnizar al titular de sus derechos ,
por los daños y perjuicios que con ello se le causen . Esta facultad se
extingue con la muerte del autor ;
</p>
        </item>
        <item eId="art6-cla1-lst1-itme">
            <num>e</num>
            <p>La de conservar y reivindicar la paternidad de la
obra ;</p>
        </item>
        <item eId="art6-cla1-lst1-itmf">
            <num>f</num>
            <p>La de oponerse al plagio de la obra ; </p>
        </item>
        <item eId="art6-cla1-lst1-itmg">
            <num>g</num>
            <p>La de exigir que su nombre o su seudónimo se
publique en cada ejemplar de la obra o se mencione en cada acto de
comunicación pública de la misma ; </p>
        </item>
        <item eId="art6-cla1-lst1-itmh">
            <num>h</num>
            <p>La de oponerse a que su nombre o su seudónimo
aparezca sobre la obra de un tercero o sobre una obra que haya sido
desfigurada ; </p>
        </item>
        <item eId="art6-cla1-lst1-itmi">
            <num>i</num>
            <p>La de salvaguardar la integridad de la obra oponiéndose a cualquier deformación, mutilación, modificación o abreviación de
la obra o de su título , incluso frente al adquirente del objeto
material de la obra ; y</p>
        </item>
        <item eId="art6-cla1-lst1-itmj">
            <num>j</num>
            <p>La de oponerse a cualquier utilización de la obra en
menoscabo de su honor o de su reputación como autor .
</p>
        </item>
    </blockList>
</content>

```

```

    </clause>
  </article>
  <article eId="art7">
    <num>Art. 7.</num>
    <clause eId="art7-cls1">
      <content>
        <p>

```

El derecho pecuniario del autor es la facultad de percibir beneficios económicos provenientes de la utilización de las obras y comprende especialmente las siguientes facultades: </p>

```

    <blockList eId="art7-cls1-lst1">
      <item eId="art7-cls1-lst1-itma">
        <num>a)</num>

```

<p>La de reproducir la obra, fijándola materialmente por cualquier procedimiento que permita comunicarla al público de una manera indirecta y durable o la obtención de copias de toda la obra o parte de ella; puede efectuarse por medios de reproducción mecánica, tales como la imprenta, la litografía, el polígrafo, el cinematógrafo, el fonógrafo, las grabaciones magnetofónicas, la fotografía y cualquier otro medio de fijación; comprende también la reproducción de improvisaciones, discursos, lectura y en general, recitaciones públicas hechas mediante la estenografía, la dactilografía y otros procedimientos análogos;</p>

```

      </item>
      <item eId="art7-cls1-lst1-itmb">
        <num>b)</num>

```

<p>La de ejecutar y representar la creación compuesta expresamente con tal propósito, comunicándola al público directa y momentáneamente, tales como la representación teatral, la ejecución musical y coreografía, la escenificación para cinematografía y televisión, y el montaje de cualesquiera otra forma de espectáculo público;

```

      </p>
      </item>
      <item eId="art7-cls1-lst1-itmc">
        <num>c)</num>

```

<p>La de difundir la obra por cualquier medio que sirva para transmitir los sonidos y las imágenes, tales como el teléfono, la radio, la televisión, el cable, el teletipo, el satélite, o por cualquier otro medio ya conocido o que se desarrolle en el futuro;</p>

```

      </item>
      <item eId="art7-cls1-lst1-itmd">
        <num>d)</num>

```

<p>La de distribución de la obra, es decir, la de poner a disposición del público los ejemplares de la obra, por medio de la



venta u otra forma de transferencia de la propiedad, pero cuando la comercialización de los ejemplares se realice mediante venta, esta facultad se extingue a partir de la primera venta, salvo las excepciones legales; conservando el titular de los derechos patrimoniales, el de autorizar o no el arrendamiento de dichos ejemplares, así como los de modificar, comunicar públicamente y reproducir la obra;

</item>

<item eId="art7-cla1-lst1-itme">

<num>e</num>

<p>La de importar, exportar o autorizar la importación o la exportación de copias de sus obras legalmente fabricadas, y la de evitar la importación o exportación de copias fabricadas en forma ilegal.

</item>

</blockList>

</content>

</clause>

</article>

</chapter>

</title>

<title eId="tlt4">

<num>TITULO QUINTO </num>

<chapter eId="tlt5-chp1">

<num>CAPITULO II</num>

<heading>DE LAS INVENCIONES</heading>

<article eId="art186">

<num>Art. 186</num>

<clause eId="art186-cla1">

<content>

<p>Los que actualmente sin autorización del titular de los derechos respectivos, se dediquen a la reproducción, venta, arrendamiento o a cualquiera otra forma de comercialización de obras audiovisuales y grabaciones sonoras reguladas en el Titulo Segundo de esta ley, gozarán del plazo de cuatro meses contados a partir de la vigencia de la misma, a efecto de obtener las autorizaciones correspondientes y vencido dicho plazo sin que éstas sean obtenidas, tales actividades se volverán ilícitas y sujetas a las sanciones pertinentes.

</content>

</clause>

</article>

</chapter>

</title>

</body>

<conclusions>

```

    <p>DADO EN EL SALÓN AZUL, PALACIO LEGISLATIVO: San Salvador , a los
    catorce días del mes de diciembre del año dos mil cinco.<eol/>
  </p>
  <p refersTo="#Zepeda">CIRO CRUZ ZEPEDA PEÑA<eol/>
</p>
  <p refersTo="#presidente">PRESIDENTE<eol/>
</p>
  <p refersTo="#Herniquez">JOSÉ MANUEL MELGAR HENRÍQUEZ<eol/>
</p>
  <p refersTo="#pv">PRIMER VICEPRESIDENTE<eol/>
</p>
  <p refersTo="#Lopez">JOSÉ FRANCISCO MERINO LÓPEZ<eol/>
</p>
  <p refersTo="#tv">TERCER VICEPRESIDENTE<eol/>
</p>
</conclusions>
</act>
</akomaNtoso>

```

doc/Act\_Salvador\_Ley\_604\_1993.xml

## A.2 Folha de estilo para a Ley 604/93

A lei salvadorenha pode ser transformada por ferramentas apropriadas para extração das informações contidas no documento. A folha de estilo XSL a seguir faz com que um documento HTML seja gerado com o título na cor azul e os artigos separados por uma linha em branco.

```

<xsl:stylesheet version="1.0" xmlns:an="http://docs.oasis-open.org/
  legaldocml/ns/akn/3.0/CSD13"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
>

<xsl:template match="an:akomaNtoso">
  <html>
    <head><title>Lei Salvadorenha</title></head>
    <body>
      <div style="background-color: wheat; padding=1em; ">
        <h1>Extraído de documento XML segundo o padrão Akoma Ntoso)</h1>
        <xsl:apply-templates/>
      </div>
    </body>
  </html>
</xsl:template>

```

```

<xsl:template match="an:eol">
<br/>
</xsl:template>

<xsl:template match="an:docTitle">
<div style="center">
<h2>
<font color="blue"><xsl:apply-templates/></font>
</h2>
</div>
</xsl:template>

<xsl:template match="an:article">
<p>
<xsl:apply-templates/>
</p>
</xsl:template>

</xsl:stylesheet>

```

doc/Act\_Salvador\_Ley\_604\_1993.xsl

A Figura A.1 mostra a saída resultante.

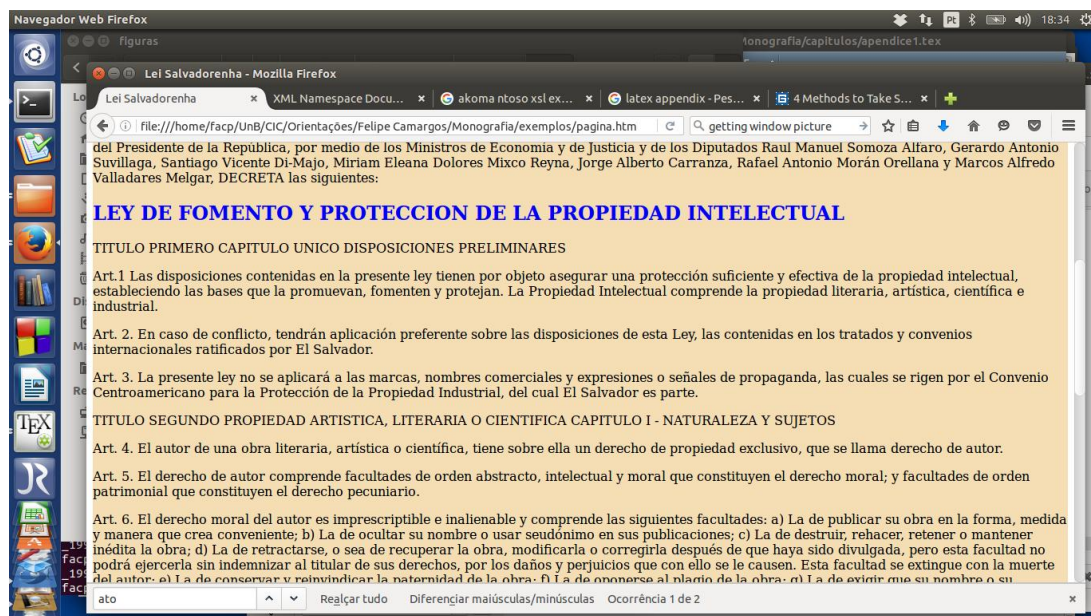


Figura A.1: Exemplo de saída produzida para a Ley 604/93 de El Salvador.